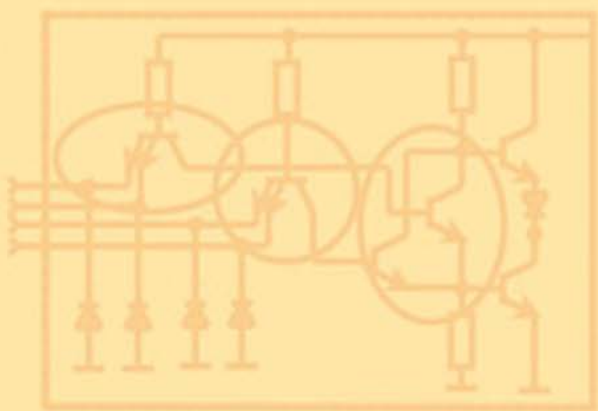


А. В. Микушин, А. М. Сажнев, В. И. Сединин

Цифровые устройства и микропроцессоры



bhv



УДК 681.3.06(075.8)
ББК 32.973.26-04я73
М59

Микушин, А. В.

М59 Цифровые устройства и микропроцессоры: учеб. пособие /
А. В. Микушин, А. М. Сажнев, В. И. Сединин. — СПб.: БХВ-Петербург,
2010. — 832 с.: ил. — (Учебная литература для вузов)

ISBN 978-5-9775-0417-1

Книга представляет собой учебник по курсу с одноименным названием, который читается авторами в течение многих лет студентам радиотехнических специальностей в Сибирском государственном университете телекоммуникаций и информатики и Новосибирском государственном техническом университете. Даны основы проектирования цифровых устройств с упором на создание принципиальных схем устройств связи. Рассмотрены вопросы аналого-цифрового преобразования и обработки сигналов, в частности, случаи изменения частоты дискретизации цифрового сигнала и узлы, позволяющие изменять эту частоту. В качестве примеров цифровых устройств рассмотрены такие современные устройства, как схемы прямого цифрового синтеза DDS, цифрового преобразования частоты вверх DUC, цифрового понижения частоты приема DDC. Изложены основы микропроцессорной техники и особенности работы микроконтроллеров на примере семейства MCS-51. Даны основы программирования для микроконтроллеров на языках С и ассемблер. Две основные части курса иллюстрируются примером разработки одного и того же устройства — часов, на цифровых микросхемах и на микроконтроллере.

Для студентов, инженеров и специалистов радиотехнических специальностей

УДК 681.3.06(075.8)
ББК 32.973.26-04я73

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Евгений Рыбаков</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Алиса Тяжбина</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Фото	<i>Кирилла Сергеева</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 29.01.10.

Формат 70×100¹/₁₆. Печать офсетная. Усл. печ. л. 67,08.

Тираж 1500 экз. Заказ №

"БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Санитарно-эпидемиологическое заключение на продукцию
№ 77.99.60.953.Д.005770.05.09 от 26.05.2009 г. выдано Федеральной службой
по надзору в сфере защиты прав потребителей и благополучия человека.

Отпечатано с готовых диапозитивов
в ГУП "Типография "Наука"
199034, Санкт-Петербург, 9 линия, 12

ISBN 978-5-9775-0417-1

© Микушин А. В., Сажнев А. М., Сединин В. И., 2010
© Оформление, издательство "БХВ-Петербург", 2010

Оглавление

Введение	1
ЧАСТЬ I. ОСНОВЫ ЦИФРОВОЙ ТЕХНИКИ	5
Глава 1. Параметры цифровых микросхем	7
Уровни логического нуля и единицы	8
Входные и выходные токи цифровых микросхем	12
Параметры, определяющие быстродействие цифровых микросхем.....	13
Описание логической функции цифровых схем	15
Итоги.....	19
Глава 2. Основные логические функции и элементы	20
Функция "НЕ", инвертор.....	20
Функция "И", логическое умножение.....	21
Функция "ИЛИ", логическое сложение.....	24
Итоги.....	26
Глава 3. Основные схемотехнические решения цифровых микросхем	27
Диодно-транзисторная логика (ДТЛ)	27
Транзисторно-транзисторная логика (ТТЛ).....	33
Логические уровни ТТЛ-микросхем	35
Семейства ТТЛ-микросхем	36
Логика на комплементарных МОП-транзисторах (КМДП).....	37
Особенности применения КМОП-микросхем.....	41
Логические уровни КМОП-микросхем.....	43
Семейства КМОП-микросхем	44
Итоги.....	45

Глава 4. Согласование цифровых микросхем между собой	46
Согласование цифровых микросхем из различных серий между собой	47
Согласование микросхем по току.....	47
Согласование микросхем с различным напряжением питания.....	49
Согласование 3- и 5-вольтовых ТТЛ-микросхем	50
Согласование 3-вольтовых ТТЛ-микросхем и 2,5-вольтовых КМОП-микросхем.....	51
Регенерация цифрового сигнала	52
Итоги.....	56
Глава 5. Арифметические основы цифровой техники	57
Системы счисления	57
Десятичная система счисления.....	59
Двоичная система счисления.....	60
Восьмеричная система счисления.....	62
Шестнадцатеричная система счисления.....	65
Преобразование чисел из одной системы счисления в другую	70
Преобразование целой части числа.....	70
Преобразование дробной части числа	74
Итоги.....	76
Глава 6. Комбинационные цифровые схемы	78
Законы алгебры логики.....	78
Закон одинарных элементов	79
Законы отрицания	81
Комбинационные законы	82
Построение цифровой схемы по произвольной таблице истинности.....	84
Декодеры	90
Десятичный дешифратор.....	90
Семисегментный дешифратор.....	93
Шифраторы	96
Мультиплексоры.....	99
Особенности построения мультиплексоров на ТТЛ-элементах.....	100
Особенности построения мультиплексоров на КМОП-элементах	102
Демультимплексоры	104
Итоги.....	106
Глава 7. Генераторы.....	107
Усилительные параметры КМОП-инвертора	108
Осцилляторные схемы	109
Мультивибраторы.....	113
Особенности кварцевой стабилизации частоты генераторов	116

Одновибраторы.....	118
Укорачивающие одновибраторы.....	118
Расширяющие одновибраторы.....	121
Применение одновибраторов.....	122
Итоги.....	123
Глава 8. Цифровые схемы последовательностного типа.....	125
Триггеры.....	125
RS-триггеры.....	126
Синхронные RS-триггеры.....	128
Статические D-триггеры.....	130
Явление метастабильности.....	133
Динамические D-триггеры.....	136
Т-триггер.....	139
JK-триггер.....	140
Регистры.....	144
Параллельные регистры.....	144
Последовательные регистры.....	147
Универсальные регистры.....	149
Счетчики.....	151
Двоичные суммирующие асинхронные счетчики.....	152
Двоичные вычитающие асинхронные счетчики.....	155
Недвоичные счетчики с обратной связью.....	158
Недвоичные счетчики с предварительной связью.....	163
Синхронные счетчики.....	167
Синхронные счетчики на регистрах сдвига.....	167
Синхронные двоичные счетчики.....	170
Итоги.....	174
Глава 9. Индикаторы.....	175
Малогабаритные лампочки накаливания.....	177
Расчет транзисторного ключа.....	177
Газоразрядные лампы.....	180
Светодиодные индикаторы.....	186
Схемы подключения светодиодных индикаторов.....	186
Виды светодиодных индикаторов.....	189
Динамическая индикация.....	191
Жидкокристаллические индикаторы.....	196
Принципы работы жидкокристаллических индикаторов.....	197
Режимы работы жидкокристаллических индикаторов.....	199
Параметры жидкокристаллических индикаторов.....	200
Формирование цветного изображения на жидкокристаллических индикаторах.....	200

Формирование управляющего напряжения для жидкокристаллического индикатора	201
Особенности динамической индикации в жидкокристаллических индикаторах	203
Итоги	206

ЧАСТЬ II. ПРИМЕРЫ РЕАЛИЗАЦИИ ЦИФРОВЫХ УСТРОЙСТВ ...207

Глава 10. Разработка цифрового устройства на примере электронных часов.....209

Разработка структурной схемы часов	209
Разработка принципиальной схемы часов	215
Разработка схемы генератора эталонных интервалов времени	216
Разработка схемы счетчика интервалов времени	220
Разработка принципиальной схемы блока индикации	222
Разработка принципиальной схемы блока коррекции времени	224
Итоги	227

Глава 11. Синхронные последовательные порты228

SSI-интерфейс (DSP-порт)	229
SPI-порт	233
I ² C-порт	240
Итоги	241

Глава 12. Синтезаторы частоты.....242

Схемы фазовой подстройки частоты	245
Схемы определения ошибки по частоте	248
Цифровой фазовый детектор	248
Фазовый компаратор	251
Умножители частоты	254
Частотные детекторы, построенные на основе ФАПЧ	256
Итоги	257

ЧАСТЬ III. СХЕМЫ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ.....259

Глава 13. Цифровая обработка сигналов261

Структурная схема цифрового устройства обработки сигнала	262
Особенности аналого-цифрового и цифроаналогового преобразования	264
Квантование аналогового сигнала по времени	265
Погрешности дискретизатора	268
Погрешность хранения	269
Погрешность выборки	270
Погрешность временного положения стробирующего импульса	274

Фильтры для устранения эффекта наложения спектров (Антиалиасинговые фильтры).....	278
Дискретизация сигнала на промежуточной частоте (субдискретизация).....	285
Статическая передаточная функция АЦП и ЦАП и погрешности по постоянному току	287
Итоги.....	296
Глава 14. Виды аналого-цифровых преобразователей.....	297
Параллельные АЦП.....	297
Последовательно-параллельные АЦП.....	299
АЦП последовательного приближения.....	302
Сигма-дельта-АЦП.....	305
Итоги.....	310
Глава 15. Основные блоки микросхем цифровой обработки сигналов	311
Двоичные сумматоры.....	311
Цифровые умножители	317
Постоянные запоминающие устройства	321
Масочное ПЗУ.....	321
Программируемые постоянные запоминающие устройства	326
ПЗУ с ультрафиолетовым стиранием.....	328
ПЗУ с электрическим стиранием информации	329
Статические оперативные запоминающие устройства (ОЗУ)	331
Цифровые фильтры	335
Схемная реализация нерекурсивного фильтра.....	341
Однородный цифровой фильтр	344
Итоги.....	347
Глава 16. Реализация передатчиков радиосигналов в цифровом виде .	348
Генераторы с цифровым управлением (NCO).....	349
Микросхемы прямого цифрового синтеза (DDS).....	354
Квадратурные модуляторы (Up converter)	355
Интерполирующие цифровые фильтры	357
Интерполирующий фильтр с конечной импульсной характеристикой.....	357
Параллельная реализация интерполирующего фильтра с конечной импульсной характеристикой	363
Интерполирующий однородный фильтр	365
Итоги.....	369
Глава 17. Реализация радиоприемников в цифровом виде.....	370
Цифровые преобразователи частоты.....	371
Цифровой квадратурный демодулятор.....	373

Децимирующие фильтры.....	374
Децимирующий фильтр с конечной импульсной характеристикой	375
Однородный децимирующий фильтр	376
Итоги.....	379

ЧАСТЬ IV. МИКРОПРОЦЕССОРЫ381

Глава 18. Принципы работы микропроцессора.....383

Виды двоичных кодов.....	384
Беззнаковые двоичные коды.....	385
Прямые знаковые двоичные коды.....	387
Знаковые обратные двоичные коды.....	388
Знаковые дополнительные двоичные коды.....	389
Представление рациональных чисел в двоичном коде с фиксированной запятой	393
Представление рациональных чисел в двоичном коде с плавающей запятой	394
Представление десятичных чисел	396
Суммирование двоично-десятичных чисел.....	397
Представление текстовых данных в памяти процессора	398
Арифметико-логические устройства	401
Классификация микропроцессоров	405
Типовые структуры операционного блока микропроцессора.....	408
Команды микропроцессора	411
Операционный блок микропроцессора	415
Блок микропрограммного управления	420
Микропрограммирование	424
Итоги.....	432

Глава 19. Принципы работы микропроцессорной системы434

Подключение внешних устройств к микропроцессору	435
Системная шина.....	439
Адресное пространство микропроцессорного устройства	441
Принципы построения параллельного порта.....	445
Параллельный порт вывода	446
Параллельный порт ввода	448
Параллельный порт ввода-вывода.....	449
Примеры использования параллельных портов	451
Ввод информации с клавиатуры	451
Обмен данными между микропроцессорами при помощи параллельных портов.....	453

Принципы построения последовательного порта	455
Синхронные последовательные порты	455
Асинхронные последовательные порты	462
Принципы построения таймеров	465
Суммирующие и вычитающие таймеры	466
Таймеры с автозагрузкой	468
Реверсивные таймеры	470
Свободнобегущие таймеры	470
Способы расширения адресного пространства микропроцессора	475
Метод страничного расширения адресного пространства	476
Метод сегментного расширения адресного пространства	479
Метод расширения адресного пространства при помощи окон	481
Динамические оперативные запоминающие устройства (ОЗУ)	482
Согласование быстродействия системной памяти и микропроцессора (кэш-память)	488
Итоги	489
Глава 20. Принципы работы микроконтроллеров	491
Семейство микроконтроллеров MCS-51	492
Архитектура микроконтроллеров MCS-51	494
Система команд микроконтроллеров MCS-51	499
Арифметические команды	500
Логические команды с байтовыми переменными	501
Команды пересылки данных	501
Битовые команды	502
Команды ветвления и передачи управления	503
Способы адресации операндов	507
Устройство параллельных портов микроконтроллеров MCS-51	510
Особенности построения памяти микроконтроллеров семейства MCS-51	518
Память программ микроконтроллеров MCS-51	519
Внешняя память данных микроконтроллеров MCS-51	521
Внутренняя память данных микроконтроллеров MCS-51	522
Регистры специальных функций	525
Внутренние таймеры микроконтроллера, особенности их применения	527
Режим 0	528
Режим 1	530
Режим 2	531
Режим 3	533
Управление таймерами/счетчиками	533
Использование таймера в качестве измерителя длительности импульсов	535
Использование таймера в качестве частотомера	537

Последовательный порт микроконтроллеров семейства MCS-51	538
Скорость приема/передачи информации через последовательный порт	540
Режим 0. Синхронный режим работы последовательного порта	542
Режим 1. Асинхронный 8-битовый режим	546
Режим 2. Асинхронный 9-битовый режим с фиксированной скоростью передачи	551
Режим 3. Асинхронный 9-битовый режим	554
Итого	554

ЧАСТЬ V. ПРОГРАММИРОВАНИЕ МИКРОКОНТРОЛЛЕРОВ.....555

Глава 21. Принципы создания программ для микроконтроллеров557

Языки программирования для микроконтроллеров.....	558
Виды программ-трансляторов.....	560
Виды компиляторов.....	560
Применение подпрограмм	561
Стек, его организация и структура.....	564
Подпрограммы-процедуры и подпрограммы-функции	565
Применение комментариев.....	567
Структурное программирование.....	569
Линейная цепочка операторов	572
Условное выполнение операторов	574
Конструкция управления циклическим выполнением оператора с проверкой условия после тела цикла	579
Структурная конструкция циклического выполнения оператора с проверкой условия до тела цикла.....	582
Понятие многофайлового и многомодульного программирования	584
Многофайловые программы	584
Многомодульные программы	590
Программа-монитор	593
Использование таймера для организации параллельных программных потоков	609
Использование прерываний для ввода информации о кратковременных сигналах и событиях, наступающих в произвольный момент времени.....	612
Итого.....	617

Глава 22. Язык программирования С-51618

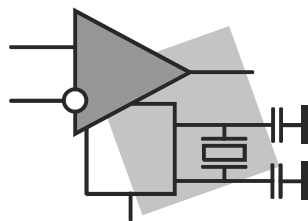
Применение С-51	619
Отладка программ.....	622
Структура программ С-51	624
Символы языка программирования С-51	626

Лексические единицы, разделители и использование пробелов	629
Идентификаторы	630
Ключевые слова	632
Константы	632
Выражения в операторах языка программирования C-51	636
Приоритеты выполнения операций	640
Операторы языка программирования C-51	641
Операторы объявления	641
Исполняемые операторы	642
Оператор присваивания	642
Условный оператор	643
Структурный оператор <i>{}</i>	644
Оператор цикла <i>for</i>	646
Оператор цикла с проверкой условия до тела цикла <i>while</i>	648
Оператор цикла с проверкой условия после тела цикла <i>do while</i>	649
Оператор <i>break</i>	650
Оператор <i>continue</i>	650
Оператор выбора <i>switch</i>	651
Оператор безусловного перехода <i>goto</i>	652
Оператор выражения	653
Оператор возвращения из подпрограммы <i>return</i>	653
Пустой оператор	654
Объявление переменных в языке программирования C-51	655
Категории типов данных	656
Целые типы данных	657
Числа с плавающей запятой	658
Переменные перечислимого типа	658
Объявление массивов в языке программирования C-51	661
Структуры	663
Поля битов	665
Объединения (смеси)	665
Объявление указателей в языке программирования C-51	667
Нетипизированные указатели	669
Память-зависимые указатели	671
Объявление новых типов переменных	671
Инициализация данных	672
Использование функций в языке программирования C-51	675
Определение функций	676
Параметры функций	679
Предварительное объявление подпрограмм	681
Вызов функций	682
Рекурсивный вызов подпрограмм	684
Подпрограммы обработки прерываний	685

Области действия переменных и подпрограмм.....	686
Итоги.....	688
Глава 23. Язык программирования ASM-51	689
Исходный текст программы на языке программирования ASM-51	691
Символы языка ASM-51	693
Идентификаторы	695
Ключевые слова.....	695
Встроенные имена.....	696
Определяемые имена	697
Числа и литеральные строки.....	697
Директивы языка программирования ASM-51	699
Управляющие команды.....	705
Реализация подпрограмм на языке ASM-51	706
Реализация подпрограмм-процедур на языке ASM-51	707
Передача переменных-параметров в подпрограмму	708
Реализация подпрограмм-функций на языке ASM-51	711
Реализация подпрограмм обработки прерываний на языке ASM-51	712
Структурное программирование на языке ASM-51	714
Многомодульные программы.....	719
Использование сегментов в языке программирования ассемблер	721
Итоги.....	728
Глава 24. Работа с интегрированной средой программирования	729
Работа с текстовым редактором интегрированной среды программирования keil-C	730
Создание программных проектов	733
Создание программного проекта в интегрированной среде программирования keil-C	735
Настройка свойств программного проекта в интегрированной среде программирования keil-C	737
Работа с программным проектом в интегрированной среде программирования keil-C	741
Трансляция программных модулей и программных проектов	741
Трансляция программных модулей.....	742
Связывание объектных модулей и получение загрузочного файла.....	749
Трансляция программных проектов.....	749
Применение интегрированной среды программирования keil-C для трансляции программного проекта	750
Отладка программ во встроенном отладчике программ	753
Способы отладки программ	753
Использование встроенного отладчика программ.....	753
Итоги.....	756

Глава 25. Пример реализации микроконтроллерного устройства	758
Структурная схема часов	758
Разработка принципиальной схемы.....	758
Разработка программы устройства	764
Разработка генератора секундных импульсов	766
Разработка подпрограммы часов	770
Разработка подпрограммы индикации	773
Разработка подпрограммы семисегментного дешифратора	777
Разработка блока коррекции часов.....	778
Итоги.....	783
ПРИЛОЖЕНИЯ	785
Приложение 1. Система команд микроконтроллеров семейства MCS-51	787
Приложение 2. Таблица ASCII-кодов	799
Список литературы	809
Предметный указатель	811

ГЛАВА 1



Параметры цифровых микросхем

Цифровые микросхемы или микросборки, их элементы или компоненты обозначаются на принципиальных схемах условно-графическим обозначением в соответствии с ГОСТ 2.743-91. Условно-графическое обозначение микросхемы имеет форму прямоугольника, к которому подводят линии выводов. Оно может содержать три поля: основное и два дополнительных, которые располагают слева и справа от основного (рис. 1.1). В первой строке основного поля помещают обозначение функции, выполняемой элементом. В последующих строках основного поля располагают информацию по ГОСТ 2.743-91.

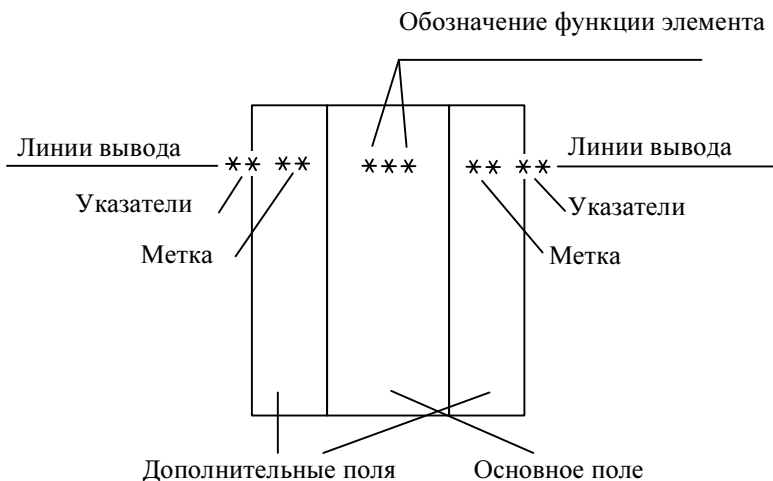


Рис. 1.1. Условно-графическое изображение цифровых микросхем

В дополнительных полях помещают информацию о назначениях выводов (метки выводов, указатели). Дополнительные поля на условно-графическом изображении цифровых микросхем могут отсутствовать. Входы на условно-графическом изображении цифровых микросхем располагают слева, а выходы — справа. Номера выводов микросхем помещают над линией вывода ближе к изображению микросхемы.

Точно так же, как и аналоговые схемы, цифровые схемы должны описываться некоторыми параметрами. Аналоговые схемы характеризуются напряжением питания, при котором они могут работать. Цифровые микросхемы тоже обладают этим параметром. В настоящее время наиболее распространены цифровые микросхемы с напряжением питания +5 В и +3,3 В, однако существуют микросхемы, способные работать в диапазоне напряжений от 2 до 6 В. Минимальное напряжение питания, при котором работают современные цифровые микросхемы составляет 0,7 В.

Уровни логического нуля и единицы

Как уже упоминалось ранее, цифровые микросхемы характеризуются тем, что могут находиться только в двух состояниях. Состояния цифровых микросхем могут быть описаны двумя цифрами: "0" и "1". При этом состояние микросхемы можно характеризовать различными параметрами. Например, током или напряжением в цепях микросхемы, открыты или закрыты транзисторы на выходе, светится или нет светодиод (если он входит в состав микросхемы).

В качестве логических состояний цифровых микросхем условились воспринимать напряжение на их входе и выходе. При этом высокое напряжение считается единицей, а низкое напряжение — нулем. В идеальном случае напряжение на выходе микросхем должно быть равным напряжению питания или нулевым потенциалом общего провода схемы.

В реальных схемах так не бывает. Даже на полностью открытом реальном транзисторе есть определенное падение напряжения. В результате на выходе цифровой микросхемы напряжение всегда будет меньше напряжения питания или больше потенциала общего провода. Поэтому в реальных схемах напряжение, меньшее заданного уровня (уровень логического нуля), считается нулем, а напряжение, большее заданного уровня (уровень логической единицы), считается единицей. Если же напряжение на выходе микросхемы будет больше уровня логического нуля, но меньше уровня логической единицы, то такое состояние микросхемы называется неопределенным.

На рис. 1.2 приведены уровни выходных логических сигналов, допустимые для цифровых ТТЛ-микросхем. ✧ *Обратите внимание*, что чем ближе вы-

ходное напряжение к напряжению питания или к напряжению общего провода схемы, тем выше КПД цифровой микросхемы.

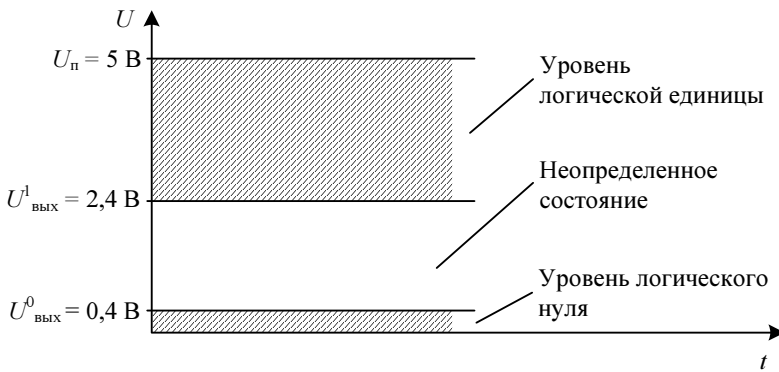


Рис. 1.2. Уровни логических сигналов на выходе цифровых ТТЛ-микросхем

В реальных схемах напряжение с выхода одной микросхемы передается на вход другой микросхемы по проводникам. В процессе передачи на эти проводники может наводиться напряжение или ток от каких-либо генераторов помех (осветительная сеть, радиопередатчики, генераторы импульсных сигналов и т. п.). В этом случае помехоустойчивость цифровых микросхем определяется максимальным напряжением помех, которое не приводит к превращению логической единицы в логический ноль, и она зависит от разности логического уровня цифровой микросхемы-источника и логического уровня микросхемы-приемника. Это напряжение можно определить из следующих выражений:

$$U_{\text{пом}}^- = \left| U_{\text{Вых1min}} - U_{\text{Вх1min}} \right|$$

То же самое относится и к помехам, превращающим логический ноль в логическую единицу. Обозначим такую помеху $U_{\text{пом}}^+$. Эта помеха определяется из следующего выражения:

$$U_{\text{пом}}^+ = \left| U_{\text{Вых0min}} - U_{\text{Вх0min}} \right|$$

Чем меньше разница между $U_{\text{Вх1min}}$ и $U_{\text{Вх0max}}$, тем большим усилением обладает цифровая микросхема. Типовое усиление ТТЛ-микросхем по напряжению K_u составляет 40 раз. Это приводит к тому, что, подав на вход этой микросхемы напряжение, на 15 мВ меньше уровня $U_{\text{пор}}$, мы воспримем его как логический ноль, и на выходе этой микросхемы получим нормальный логический уровень. При подаче на вход ТТЛ-микросхемы напряжения, на 15 мВ

большого уровня $U_{\text{пор}}$, это напряжение будет восприниматься как логическая единица. Данное рассуждение иллюстрируется следующей формулой:

$$U_{\text{вых}} = K_u \times U_{\text{вх}} = 15 \times 40 \text{ мВ} = 0,6 \text{ В}$$

Если теперь из порогового напряжения вычесть 0,6 В, то мы получим 0,8 В — уровень нормального логического нуля. А если к нему прибавить 0,6 В, то получим 2 В — минимальный уровень логической единицы. *Обратите внимание, что описанная ситуация предполагает отсутствие помех.*

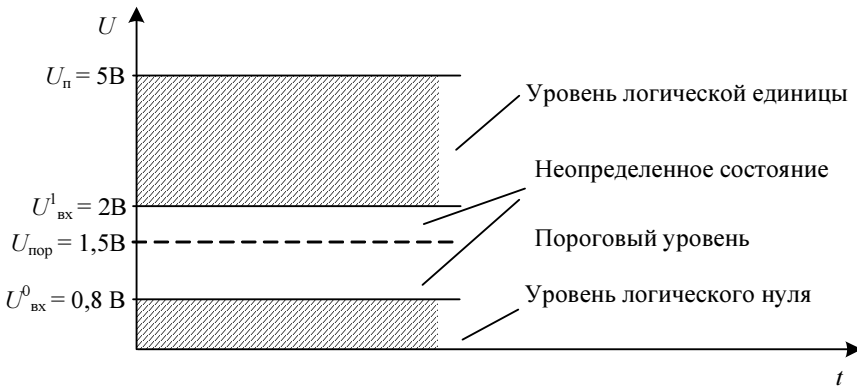


Рис. 1.3. Уровни логических сигналов на входе цифровых ТТЛ-микросхем

На рис. 1.3 приведены входные уровни логического нуля и логической единицы ТТЛ-микросхем. На этом же рисунке представлен пороговый уровень входного сигнала (граница уровней логического нуля и единицы). Этот уровень немного смещен от середины защитного интервала, т. к. переходная характеристика биполярных транзисторов нелинейна.

Обратите внимание, что входные уровни логических сигналов отличаются от выходных. Это связано с тем, что микросхемы могут работать при воздействии неблагоприятных факторов, таких как пониженная температура, старение микросхем, воздействие радиации. На цепи питания и общего провода могут наводиться помехи. Поэтому производители микросхем гарантируют срабатывание микросхем с некоторым запасом. Например, фирма Texas Instruments объявляет для своих микросхем входной уровень единицы — 2 В, а уровень нуля — 0,8 В.

Теперь можно определить уровень помехи, который не будет восприниматься цифровой микросхемой. В качестве примера выберем ТТЛ-микросхемы. Мы уже знаем, что на выходе цифровой ТТЛ-микросхемы уровень логической единицы не может быть меньше 2,4 В, а уровень логического нуля не может быть больше 0,4 В. То есть напряжение помехи, гарантированное производи-

телем, будет $U_{\text{пом}} = 2,4 - 2 = 0,8 - 0,4 = 0,4$ В. В реальности, даже при наведении на вход ТТЛ-микросхемы помехи, напряжением $U_{\text{пом}} = U^1 - U_{\text{пор}} = 2,4 - (1,5 + 0,04) = 0,86$ вольт, искажения цифровой информации не произойдет.

А что же произойдет, если напряжение на входе цифровой микросхемы будет близко к порогу, разделяющему уровень логического нуля и логической единицы? В этом случае микросхема перейдет в активный (усилительный) режим работы, и оба выходных транзистора могут оказаться открытыми. В результате микросхема может выйти из строя из-за перегрева. Поэтому *входы цифровых (особенно КМОП) микросхем ни в коем случае не должны быть оставлены неподключенными!* Если часть элементов цифровой микросхемы не используется, то их входы должны быть подключены к источнику питания или общему проводу схемы. Конкретная точка подключения неиспользуемого входа микросхемы будет подробно обсуждаться позднее. Соотношение уровней логических сигналов на выходе и входе логических микросхем приведено на рис. 1.4.

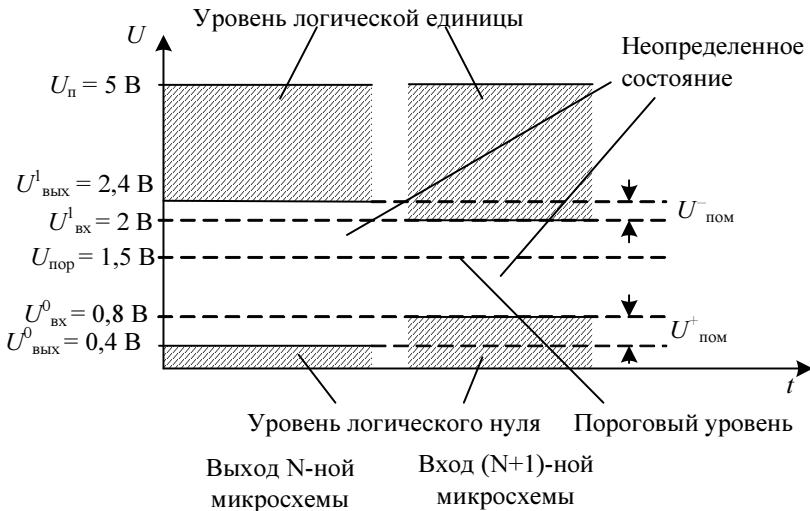


Рис. 1.4. Соотношение уровней логических сигналов на выходе и входе цифровых ТТЛ-микросхем

В заключение следует обратить внимание на то, что конкретное значение порога переключения для различных экземпляров и серий микросхем может изменяться в некоторых пределах. Это еще одна причина, по которой нельзя подавать на вход логических микросхем напряжение в пределах неопределенного состояния или оставлять входы микросхем неподключенными.

Входные и выходные токи цифровых микросхем

Еще один важный параметр любой микросхемы — это предельно допустимый выходной ток. Для цифровых микросхем есть два различных значения выходного тока: ток единицы (высокого потенциала) и ток нуля (низкого потенциала). В цифровых микросхемах эти значения могут быть различными. Путь протекания тока единицы в цифровых микросхемах показан на рис. 1.5.

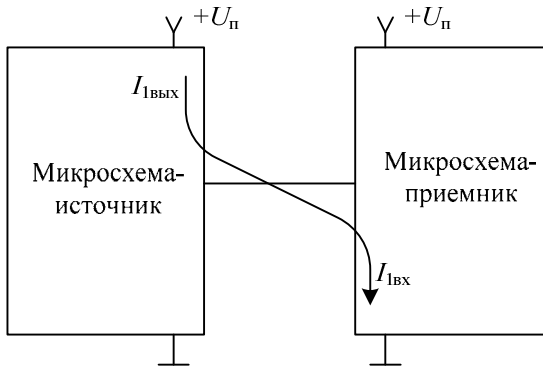


Рис. 1.5. Путь протекания выходного тока единицы в цифровых микросхемах

На этом рисунке видно, что в простейшем случае выходной ток цифровой микросхемы $I_{\text{вых}}$ (вытекающий ток) совпадает с входным током единицы $I_{\text{вх}}$ нагрузочной цифровой микросхемы (микросхемы-приемника). Часто требуется подавать сигнал с выхода одной микросхемы на несколько других микросхем. В этом случае выходной ток микросхемы будет определяться как сумма входных токов микросхем-приемников. Количество однотипных микросхем, которые могут быть одновременно подключены к выходу микросхемы, определяет *предельную нагрузочную способность микросхемы*.

Путь протекания выходного тока нуля $I_{0\text{вых}}$ (втекающий ток) показан на рис. 1.6. В этом случае выходной ток нуля микросхемы-источника $I_{0\text{вых}}$ тоже определяется суммой входных токов нуля $I_{0\text{вх}}$ микросхем-приемников, подключенных к ее выходу.

Для того чтобы выход цифровой микросхемы мог нагружаться на входы нескольких микросхем-приемников, их входной ток должен быть меньше выходного тока микросхемы-источника. Для ТТЛ-микросхем нагрузочная способность составляет обычно 10. Для КМОП-микросхем она может достигать 100, т. е. на выход одной КМОП-микросхемы можно нагружать до сотни входов других КМОП-микросхем.

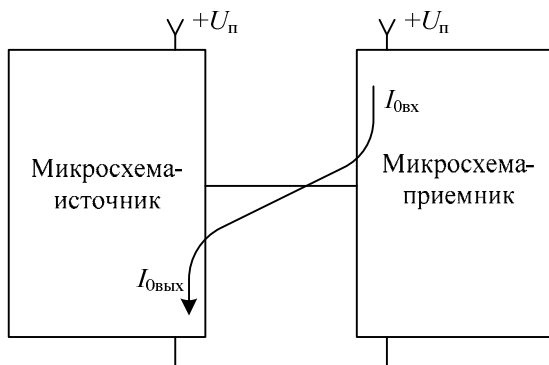


Рис. 1.6. Путь протекания выходного тока нуля в цифровых микросхемах

Параметры, определяющие быстродействие цифровых микросхем

Быстродействие цифровых микросхем определяется скоростями их перехода из одного состояния в другое. При этом быстродействие цифровой микросхемы определяется временем задержки выходного сигнала относительно входного. Не следует путать это время с длительностью фронта выходного импульса цифровой микросхемы.

В общем случае длительность переднего (rising — нарастающего) и заднего (falling — спадающего) фронтов цифрового сигнала не совпадает. Длительность фронта определяется как время нарастания выходного сигнала от напряжения $0,1 \times U$ до напряжения $0,9 \times U$, где U — это разность напряжений между уровнем логической единицы и уровнем логического нуля. На рис. 1.7 длительность переднего (rising — нарастающий) фронта обозначена как $t_{\text{ф}01}$, а длительность заднего (falling — спадающий) фронта — как $t_{\text{ф}10}$.

Время задержки выходного сигнала относительно входного обычно больше длительности фронта выходного сигнала, и именно этот параметр приводится в качестве характеристики цифровой микросхемы, определяющей ее быстродействие. Это время может быть измерено по точке пересечения входным и выходным сигналами порогового уровня. В цифровых микросхемах время задержки переднего фронта и время задержки заднего фронта обычно не совпадают. Времена задержки t^{01} и t^{10} показаны на временной диаграмме входного и выходного сигнала цифровой микросхемы, приведенной на рис. 1.8.

Тем не менее, для того, чтобы можно было сравнивать цифровые микросхемы между собой, часто пользуются усредненным временем задержки сигнала

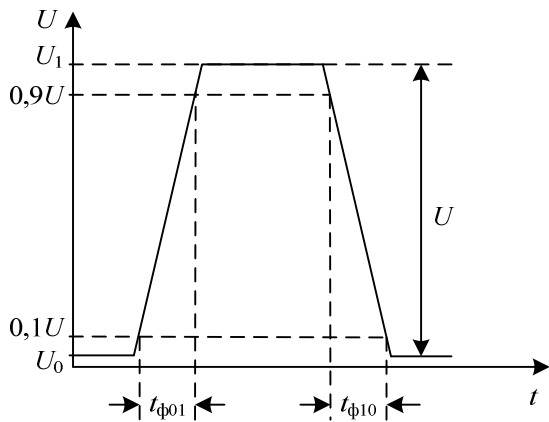


Рис. 1.7. Определение длительности переднего и заднего фронта выходного импульса

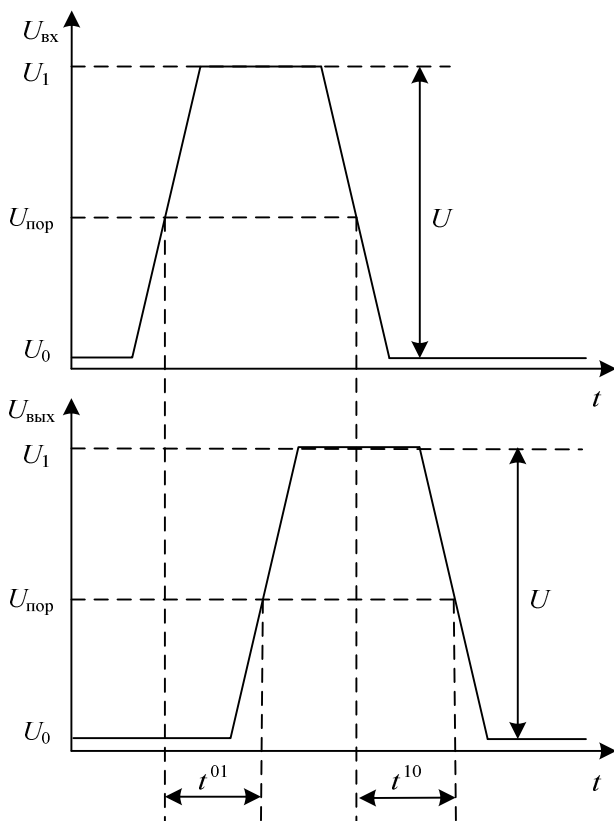


Рис. 1.8. Определение времени задержки цифровой микросхемы

цифровой микросхемой. Этот параметр определяется как половина суммы времен задержек t^{01} и t^{10} :

$$t_{\text{зад}} = \frac{t^{01} + t^{10}}{2}.$$

Описание логической функции цифровых схем

Как уже упоминалось ранее, особенностью использования цифровой техники является то, что при разработке схем можно отвлечься от особенностей практической реализации цифровых микросхем, а также от влияния конкретных значений выходного напряжения и токов нагрузки. При этом все внимание можно уделить разработке целевой логической функции, реализуемой цифровой схемой. Для этого входные и выходные сигналы цифровой схемы можно задать цифрами "0" и "1". При этом неважно, с использованием какой схемотехники (ТТЛ, КМОП или BiCMOS) будут реализованы сами логические элементы. Выбор конкретной схемотехники влияет только на эксплуатационные характеристики реализуемой аппаратуры, но совершенно не отразится на виде логической функции. Именно поэтому логические элементы, выполненные с использованием различных серий интегральных цифровых микросхем, изображаются на принципиальных схемах совершенно одинаково.

Выходные сигналы цифровых микросхем необходимо рассматривать в статическом режиме, когда все переходные процессы уже закончились, благодаря этому исключается влияние задержек распространения сигналов.

В простейших цифровых схемах выходные сигналы зависят только от входных сигналов, и не зависят от их значений в предыдущие моменты времени. Такие цифровые устройства получили название комбинационных цифровых устройств. Обычно такие устройства описываются при помощи таблицы истинности.

Процесс разработки цифрового устройства достаточно сильно алгоритмизирован. Это позволяет значительно сократить время разработки устройства и снизить вероятность возникновения ошибки на этапе проектирования устройства.

Разработка цифрового устройства разбивается на пять этапов:

1. Создание таблицы истинности.
2. Запись логического выражения.
3. Минимизация полученного логического выражения.

4. Создание схемы по минимизированному логическому выражению.
5. Реализация полученной схемы в заданном наборе микросхем.

Таблица истинности — это совокупность всех возможных комбинаций логических сигналов на входе цифрового устройства и значений выходных сигналов для каждой комбинации. Для того чтобы не пропустить ни одной комбинации входных сигналов, их обычно записывают в виде двоичного кода. Пример таблицы истинности приведен в табл. 1.1.

Таблица 1.1. Таблица истинности цифровой микросхемы

№ комбинации	вх1	вх2	вх3	вых1	вых2
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

Для описания принципов работы комбинационной цифровой схемы достаточно таблицы истинности. Этой же таблицы достаточно для создания ее принципиальной схемы.

Рассмотрим несколько примеров описания цифровой схемы с помощью таблицы истинности. Пусть требуется преобразовать двоично-десятичный код в десятичный. Поставим в соответствие каждой комбинации двоично-десятичного кода соответствующее десятичное значение (для отображения десятичных цифр необходимо использовать десять выходов схемы). Получившаяся таблица истинности содержится в табл. 1.2.

Составим еще одну таблицу истинности. На этот раз в качестве исходной постановки задачи используем временные диаграммы. В микропроцессорной технике достаточно часто приходится сталкиваться с задачей преобразования временных диаграмм системной шины (например, в процессорах фирмы Freescale, бывшая Motorola) во временные диаграммы системной шины, использующиеся в процессорах фирмы Intel. Эти временные диаграммы приведены на рис. 1.9. Сигналы DS и R/W соответствуют временным диаграммам системной шины процессоров фирмы Freescale, а инверсные сигналы RD

Таблица 1.2. Таблица истинности десятичного дешифратора

№ комбинации	Входы				Выходы									
	8	4	2	1	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0	1	0	0	0	0	0	0
4	0	1	0	0	0	0	0	0	1	0	0	0	0	0
5	0	1	0	1	0	0	0	0	0	1	0	0	0	0
6	0	1	1	0	0	0	0	0	0	0	1	0	0	0
7	0	1	1	1	0	0	0	0	0	0	0	1	0	0
8	1	0	0	0	0	0	0	0	0	0	0	0	1	0
9	1	0	0	1	0	0	0	0	0	0	0	0	0	1

и WR — временным диаграммам системной шины процессоров, совместимых с процессорами фирмы Intel.

Преобразуем приведенные временные диаграммы в таблицу истинности. Для этого проведем вертикальное сечение этих диаграмм. Видим, что в начале временных диаграмм сигналы обеих системных шин соответствуют единичным потенциалам ($DS = 1$, $R/W = 1$, $WR = 1$, $RD = 1$). Записываем это состояние в таблицу истинности. Оно соответствует отсутствию каких-либо операций в системной шине. Следующая комбинация соответствует подготовке к операции записи в системной шине процессоров фирмы Freescale ($DS = 1$, $R/W = 0$). В системной шине процессоров фирмы Intel нет аналога, поэтому заменим комбинацией отсутствия операций $WR = 1$, $RD = 1$ (то же самое мы видим и по сигналам, приведенным на временных диаграммах). Аналогичным образом запишем в таблицу истинности потенциалы сигналов, соответствующие операциям записи и чтения системной памяти. В результате описанных действий мы получили таблицу истинности для схемы преобразования сигналов системных шин двух разных микропроцессоров (табл. 1.3).

Теперь эта таблица истинности может быть использована для записи логического выражения и создания принципиальной схемы перекодировки сигналов записи/чтения, использующихся в системных шинах разных производителей микросхем.

Таблица 1.3. Таблица истинности схемы перекодировки

№ комбинации	Входы		Выходы		Операция
	DS	R/W	WR	RD	
0	0	0	0	1	запись
1	0	1	1	0	чтение
2	1	0	1	1	—
3	1	1	1	1	—

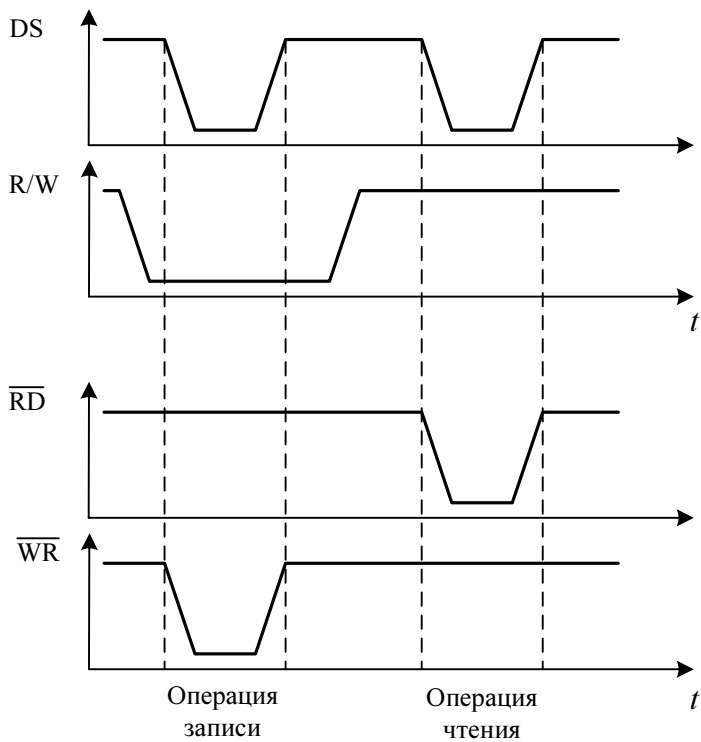
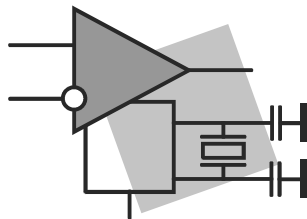


Рис. 1.9. Пример временных диаграмм цифрового устройства

Итоги

На этом можно закончить рассмотрение основных параметров цифровых микросхем. Полученных данных достаточно для того, чтобы начать работу с цифровыми микросхемами. Хотелось бы обратить внимание, что в последующих главах мы не будем касаться рассмотренных в данной главе вопросов. Однако при реальном проектировании цифровых устройств необходимо всегда учитывать рассмотренные параметры микросхем.

ГЛАВА 2



Основные логические функции и элементы

Практически все цифровые устройства без памяти (комбинаторные устройства) могут быть построены на основе трех простейших логических элементов. Все более сложные схемы реализуются из этих простейших логических элементов как из кубиков. Методы синтеза цифровых схем мы рассмотрим позднее, а пока остановимся подробнее на принципах работы и методах описания простейших логических элементов.

Функция "НЕ", инвертор

Простейшим логическим элементом является инвертор, который просто изменяет значение входного сигнала на прямо противоположное значение. Его функция записывается в следующем виде:

$$F(x) = \bar{x},$$

где черта над входным значением обозначает изменение его на противоположное. То же самое действие можно записать при помощи таблицы истинности, приведенной в табл. 2.1. Так как вход у этого логического элемента лишь один, его таблица истинности состоит только из двух строк.

Таблица 2.1. Таблица истинности логического инвертора

x	F
0	1
1	0

В качестве инвертора в простейшем случае можно использовать обычный усилитель с транзистором, включенным по схеме с общим эмиттером или

исток. Схема усилителя, выполненная на биполярном п-р-п транзисторе и позволяющая реализовать функцию логического инвертирования, приведена на рис. 2.1.

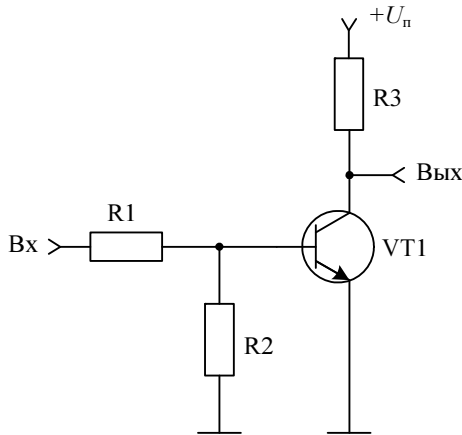


Рис. 2.1. Схема, позволяющая реализовать функцию логического инвертирования

Схемы инверторов могут обладать различным временем распространения сигнала и могут работать на различные виды нагрузки. Они могут быть выполнены на одном или на нескольких транзисторах, но независимо от схемы и ее параметров они осуществляют одну и ту же логическую функцию.

Для того чтобы особенности включения транзисторов не затеняли выполняемую функцию, для цифровых микросхем введены специальные условно-графические обозначения. Условно-графическое изображение инвертора приведено на рис. 2.2.

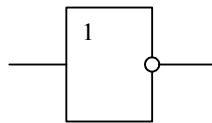


Рис. 2.2. Условно-графическое изображение логического инвертора

Функция "И", логическое умножение

Следующим простейшим логическим элементом является схема, реализующая операцию логического умножения "И":

$$F(x_1, x_2) = x_1 \wedge x_2,$$

где символ \wedge обозначает функцию логического умножения (конъюнкцию). Иногда эта же функция записывается в другом виде:

$$F(x_1, x_2) = x_1 \wedge x_2 = x_1 \cdot x_2 = x_1 \& x_2.$$

То же самое действие можно записать при помощи таблицы истинности, приведенной в табл. 2.2. В формуле, приведенной выше, использовано два аргумента. Поэтому элемент, выполняющий эту функцию, имеет два входа. Такой элемент обозначается "2И". Для элемента "2И" таблица истинности будет состоять из четырех строк. Количество строк таблицы истинности можно определить по формуле $N = 2^n$, где N — это количество строк в таблице истинности, а n — количество входов логического элемента. В нашем случае $N = 2^2 = 4$.

Таблица 2.2. Таблица истинности схемы, выполняющей логическую функцию "2И"

x_1	x_2	F
0	0	0
0	1	0
1	0	0
1	1	1

Как видно из приведенной таблицы истинности, активный сигнал на выходе этого логического элемента появляется только тогда, когда и на входе x_1 и на входе x_2 будут присутствовать логические единицы. То есть этот логический элемент действительно реализует операцию "И".

Условно-графическое изображение схемы, выполняющей логическую функцию "2И", на принципиальных схемах приведено на рис. 2.3, и с этого момента схемы, выполняющие функцию "И", будут приводиться именно в таком виде. Это изображение не зависит от конкретной принципиальной схемы устройства, реализующей функцию логического умножения.

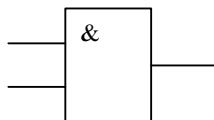


Рис. 2.3. Условно-графическое изображение схемы, выполняющей логическую функцию "2И"

Проще всего понять, как работает такой элемент при помощи схемы, построенной на идеализированных ключах с электронным управлением, как это показано на рис. 2.4. В приведенной схеме ток будет протекать только тогда, когда оба ключа будут замкнуты, а значит, единичный уровень на выходе схемы появится только при подаче на ее вход двух логических единиц.

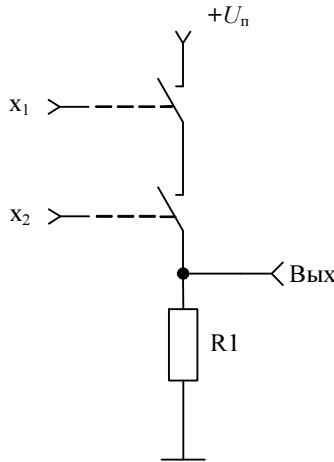


Рис. 2.4. Эквивалентная схема, реализующая логическую функцию "ИИ"

Аналогично описывается и функция логического умножения трех переменных:

$$F(x_1, x_2, x_3) = x_1 \wedge x_2 \wedge x_3.$$

Ее таблица истинности будет содержать уже восемь строк ($2^3 = 8$). Таблица истинности трехвходовой схемы логического умножения "ЗИ" приведена в табл. 2.3, а условно-графическое изображение этого логического элемента на рис. 2.5. При этом в эквивалентной схеме, реализующей таблицу истинности и построенной по принципу схемы, приведенной на рис. 2.4, добавляется третий ключ.

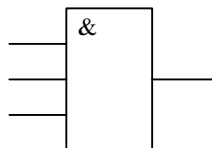


Рис. 2.5. Условно-графическое изображение схемы, выполняющей логическую функцию "ЗИ"