

ФГБОУ ВПО «Воронежский государственный
технический университет»

А.А. Пирогов

**ПРОЕКТИРОВАНИЕ ИНТЕГРАЛЬНЫХ
СХЕМ И ИХ ФУНКЦИОНАЛЬНЫХ УЗЛОВ**

Утверждено Редакционно-издательским советом университета
в качестве учебного пособия

Воронеж 2014

Пирогов А.А. Проектирование интегральных схем и их функциональных узлов: учеб. пособие [Электронный ресурс]. – Электрон. текстовые и граф. данные (810 Кб) / А.А. Пирогов. – Воронеж: ФГБОУ ВПО «Воронежский государственный технический университет», 2014. – 1 электрон. опт. диск (CD-ROM) : цв. – Систем. требования : ПК 500 и выше ; 256 Мб ОЗУ ; Windows XP ; SVGA с разрешением 1024x768 ; Adobe Acrobat ; CD-ROM дисковод ; мышь. – Загл. с экрана. – Диск и сопровод. материал помещены в контейнер 12x14 см.

В учебном пособии рассматриваются основные этапы проектирования интегральных схем, структуры и принципы работы применяемых функциональных узлов, а также средства их проектирования. Отдельный раздел посвящен системам на кристалле, элементам их структуры и маршруту проектирования.

Издание соответствует требованиям Федерального государственного образовательного стандарта высшего профессионального образования по направлению 12.03.01 «Приборостроение» (профиль «Приборостроение»), дисциплине «Специализированные БИС и устройства функциональной электроники в приборостроении»

Табл. 3. Ил. 49. Библиогр.: 4 назв.

Рецензенты: кафедра основ радиотехники и электроники Воронежского института ФСИИ (нач. кафедры канд. техн. наук, доц. Р.Н. Андреев); д-р техн. наук, проф. В.М. Питолин

© Пирогов А.А., 2014

© Оформление. ФГБОУ ВПО «Воронежский государственный технический университет», 2014

ВВЕДЕНИЕ

Внедрение полупроводников в радиоэлектронное производство уменьшило габариты устройств и потребляемую ими мощность, и как следствие повысило их быстродействие и надежность. Радиоэлектронные средства (РЭС), элементной базой которых служили отдельные полупроводниковые транзисторы, получили название РЭС второго поколения.

Развитие полупроводниковой электроники способствовало зарождению интегральной технологии, которая позволила изготавливать в одном кристалле полупроводника несколько транзисторов, объединенных в электрическую схему. Так появились функциональные узлы, ставшие элементной базой РЭС, третьего поколения.

Дальнейшее развитие интегральной технологии привело к появлению больших (БИС) и сверхбольших (СБИС) интегральных схем, на базе которых создаются цифровые устройства четвертого поколения. На одном кристалле полупроводника размещается несколько функциональных узлов, объединенных в схему - функциональный блок. Использование в устройствах БИС и СБИС способствует дальнейшему уменьшению габаритов и энергопотребления, повышению быстродействия и надежности устройств.

1. ИНТЕГРАЛЬНЫЕ УСТРОЙСТВА МИКРОЭЛЕКТРОНИКИ

1.1. Основные определения и понятия

Электрический сигнал - это электрическая величина (например, напряжение, ток, мощность), изменяющаяся со временем. Вся электроника в основном работает с электрическими сигналами, хотя сейчас все больше используются световые сигналы, которые представляют собой изменяющуюся во времени интенсивность света.

Аналоговый сигнал - это сигнал, который может принимать любые значения в определенных пределах. Устройства, работающие только с аналоговыми сигналами, называются аналоговыми устройствами.

Цифровой сигнал - это сигнал, который может принимать только два значения, причем разрешены некоторые отклонения от этих значений. Например, напряжение может принимать два значения: от 0 до 0,5В (уровень нуля) или от 2,5 до 5В (уровень единицы). Устройства, работающие исключительно с цифровыми сигналами, называются цифровыми устройствами.

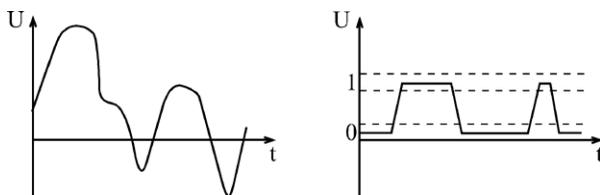


Рис. 1.1. Электрические сигналы: аналоговый (слева) и цифровой (справа)

Можно сказать, что в природе практически все сигналы - аналоговые, то есть они изменяются непрерывно в каких-то пределах. Именно поэтому первые электронные устройства были аналоговыми. Они преобразовывали физические величины в пропорциональные им напряжение или ток, производили над

ними операции и затем выполняли обратные преобразования в физические величины [1,3].

Однако аналоговые сигналы и работающая с ними аналоговая электроника имеют большие недостатки, связанные именно с природой аналоговых сигналов. Дело в том, что аналоговые сигналы чувствительны к действию всевозможных паразитных сигналов - шумов, наводок, помех.

Шум - это внутренние хаотические слабые сигналы любого электронного устройства (микрофона, транзистора, резистора ит.д.). Наводки и помехи - это сигналы, приходящие на электронную систему извне и искажающие полезный сигнал (например, электромагнитные излучения от радиопередатчиков или от трансформаторов)

В отличие от аналоговых, цифровые сигналы, имеющие всего два разрешенных значения, защищены от действия шумов, наводок и помех гораздо лучше. Небольшие отклонения от разрешенных значений никак не искажают цифровой сигнал, так как всегда существуют зоны допустимых отклонений. Именно поэтому цифровые сигналы допускают, гораздо более сложную и многоступенчатую обработку, гораздо более длительное хранение без потерь и гораздо более качественную передачу, чем аналоговые.

Однако у цифровых сигналов есть и крупный недостаток. Дело в том, что на каждом из своих разрешенных уровней цифровой сигнал должен оставаться хотя бы в течение какого-то минимального временного интервала, иначе его невозможно будет распознать. А аналоговый сигнал может принимать любое свое значение бесконечно малое время. Кроме того, цифровой сигнал передает информацию только двумя уровнями и изменением одного своего уровня на другой, а аналоговый - еще и каждым текущим значением своего уровня, то есть он более емкий с точки зрения передачи информации. Поэтому для передачи того объема информации, который содержится в одном аналоговом сигнале, чаще всего приходится использовать несколько цифровых.

Элементную базу цифровых устройств (ЦУ) составляют интегральные схемы (ИС). ИС постоянно совершенствуются и усложняются. Характеристикой сложности ИС является уровень интеграции, оцениваемый либо числом базовых логических элементов, либо числом транзисторов, которые размещены на кристалле.

Различия в уровне интеграции делят ИС на несколько категорий: МИС, СИС, БИС, СБИС (соответственно малые, средние, большие и сверхбольшие ИС). Практическое использование находят все категории, однако с течением времени все большую долю используемых микросхем составляют схемы высокого уровня интеграции.

МИС реализуют простейшие логические преобразования и обладают универсальностью - даже с помощью одного типа логического элемента (например, «И-НЕ») можно построить любое ЦУ. В виде СИС выпускаются в готовом виде такие схемы, как малоразрядные регистры, счетчики, дешифраторы, сумматоры и т. п.

С появлением БИС и СБИС схемы с тысячами и миллионами логических элементов стали размещаться на одном кристалле. При этом проблема снижения универсальности для ИС с жесткой структурой обострилась бы чрезвычайно - пришлось бы производить огромное число типов ИС при снижении объема производства каждого из типов, что непомерно увеличило бы их стоимость, т. к. высокие затраты на проектирование БИС/СБИС относились бы к небольшому объему их выпуска.

Выход из возникшего противоречия был найден на пути переноса специализации микросхем в область программирования. Появились микропроцессоры и БИС/СБИС с программируемой структурой.

Микропроцессор способен выполнять команды, входящие в его систему команд. Меняя последовательность и состав команд, можно решать различные задачи на одном и том же микропроцессоре. В виде БИС/СБИС с программируемой структурой потребителю предлагается кристалл, содержащий множество логических блоков, межсоединения для которых назначает

инженер-системотехник, программируя структуру ИС соответственно своему проекту. Разработан целый спектр методов программирования связей между блоками и элементами кристалла. Далее это будет рассмотрено более подробно.

Два указанных метода имеют большие различия. Микропроцессоры реализуют последовательную обработку информации, выполняя большое число отдельных действий, соответствующих командам, что может не обеспечить требуемого быстродействия. В БИС/СБИС с программируемой структурой обработка информации возможна без разбиения этого процесса на последовательно выполняемые элементарные действия. Задача может решаться «целиком», ее характер определяет структуру устройства. Таким образом, БИС/СБИС с программируемой структурой могут быстрее решать задачи, сложность которых ограничена уровнем интеграции микросхем, а микропроцессорные средства - задачи неограниченной сложности, но с меньшим быстродействием.

Современные кристаллы высшего уровня интеграции содержат одновременно и микропроцессоры, и большие массивы программируемой логики, обладая в силу этого большими функциональными возможностями. Подобная структура свойственна микросхемам класса «система на кристалле» СНК.

С ростом уровня интеграции ИС в проектировании на их основе все больше усиливается аспект, который можно назвать интерфейсным проектированием. Задачей разработки становится составление блоков из субблоков стандартного вида путем правильного их соединения и привлечения систем автоматизированного проектирования (САПР) для создания сложных систем.

ИС широкого применения изготавливаются по технологиям КМОП, ТТЛШ и др. Технологии КМОП и ТТЛШ стали основными. Современные элементы КМОП обладают рядом параметров (малая потребляемая мощность, особенно при невысоких частотах переключения, высокая помехоустойчивость, высокое быстродействие ит.п.). Эти элементы доминируют в схемах внутренних областей БИС/СБИС. За ТТЛШ осталась пока об-

ласть периферийных схем, где требуется передача сигналов по внешним цепям [3].

1.2. Функции цифровых устройств

Любое цифровое устройство от самого простейшего до самого сложного всегда действует по одному и тому же принципу (рис. 1.2). Оно принимает входные сигналы, выполняет их обработку, передачу, хранение и выдает выходные сигналы. При этом совсем не обязательно любое изменение входных сигналов приводит к немедленному и однозначному изменению выходных сигналов. Реакция устройства может быть очень сложной, отложенной по времени.

В качестве входных сигналов нашего устройства могут выступать сигналы с выходов других цифровых устройств, с тумблеров и клавиш или с датчиков физических величин. Причем в последнем случае, как правило, необходимо преобразование аналоговых сигналов с датчиков в потоки цифровых кодов (рис. 1.3) с помощью аналого-цифровых преобразователей (АЦП).

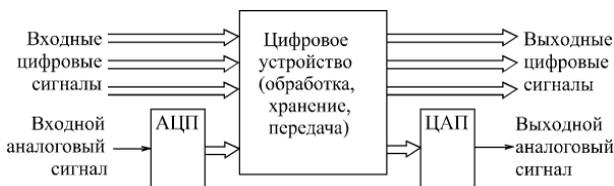


Рис. 1.2. Включение цифрового устройства

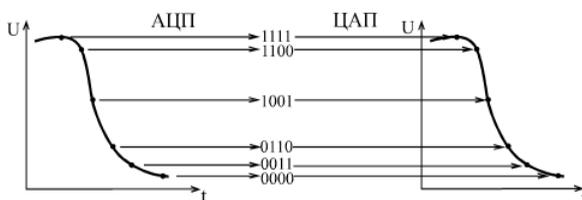


Рис. 1.3. Аналого-цифровое и цифро-аналоговое преобразование

Выходные сигналы цифрового устройства могут предназначаться для подачи на другие цифровые устройства, для индикации (на экране монитора, на цифровом индикаторе и т.д.), а также для формирования физических величин. Причем в последнем случае необходимо преобразовывать потоки кодов с цифрового устройства в непрерывные (аналоговые) сигналы (рис. 1.3) с помощью цифро-аналоговых преобразователей (ЦАП) и в физические величины.

Одно цифровое устройство может состоять из нескольких более простых цифровых устройств. Часто эти составные элементы называют блоками, модулями, узлами, частями. Если объединяются несколько сложных цифровых устройств, то говорят уже о цифровых системах, комплексах, установках.

Связь между входными и выходными сигналами может быть жесткой, неизменной или гибко изменяемой (программируемой). То есть цифровое устройство может работать по жесткому, раз и навсегда установленному алгоритму или по алгоритму программируемому. Как правило, при этом выполняется один очень простой принцип: чем больше возможностей для изменения связи входных и выходных сигналов, чем больше возможностей изменения алгоритма работы, тем цифровое устройство будет медленнее.

Значительное число задач может быть решено как чисто аппаратным путем (с помощью устройств на жесткой логике), так и программно-аппаратным путем (с помощью программируемых устройств). В таких случаях обращают внимание какие характеристики устройства являются самыми важными: скорость работы, стоимость, гибкость, простота проектирования и т.д.[3].

1.3. Простейшие и сложные логические элементы

Логические элементы (вентили) - это наиболее простые цифровые микросхемы. Как правило, в одном корпусе микросхемы может располагаться от одного до шести одинаковых ло-

гических элементов. Иногда в одном корпусе могут располагаться и разные логические элементы.

Обычно каждый логический элемент имеет несколько входов (от одного до двенадцати) и один выход. При этом связь между выходным сигналом и входными сигналами (таблица истинности) предельно проста. Каждой комбинации входных сигналов элемента соответствует уровень нуля или единицы на его выходе. Никакой внутренней памяти у логических элементов нет, поэтому они относятся к группе так называемых комбинационных микросхем. Но в отличие от более сложных комбинационных микросхем, логические элементы имеют входы, которые не могут быть разделены на группы, различающиеся по выполняемым ими функциям.

Самый простой логический элемент – это инвертор (логический элемент «НЕ»). Инвертор выполняет простейшую логическую функцию - инвертирование, то есть изменение уровня входного сигнала на противоположный. Он имеет всего один вход и один выход. На рис. 1.4 показаны условные обозначения инвертора, принятые у нас и за рубежом, а в табл. 1.1 представлена таблица истинности инвертора.

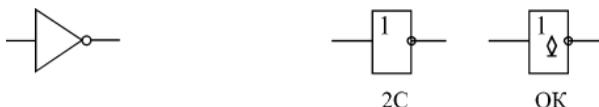


Рис. 1.4. Условные обозначения инверторов: зарубежные (слева) и отечественные (справа)

Таблица 1.1

Таблица истинности инвертора

Вход	Выход
0	1
1	0

Две основные области применения инверторов - это изменение полярности сигнала и изменение полярности фронта

сигнала (рис. 1.5). То есть из положительного входного сигнала инвертор делает отрицательный выходной сигнал и наоборот, а из положительного фронта входного сигнала - отрицательный фронт выходного сигнала и наоборот. Еще одно важное применение инвертора — буферирование сигнала (с инверсией), то есть увеличение нагрузочной способности сигнала [3].



Рис. 1.5. Инверсия полярности сигнала и инверсия полярности фронта сигнала

Элементы «И, И-НЕ, ИЛИ, ИЛИ-НЕ» - это элементы, выполняющие простейшие логические функции. Объединяет все эти элементы то, что у них есть несколько равноправных входов (от 2 до 12) и один выход, сигнал на котором определяется комбинацией входных сигналов.

Название самих функций «И» и «ИЛИ» говорит о том, при каком условии на входах появляется сигнал на выходе. При этом важно помнить, что речь в данном случае идет о положительной логике, о положительных, единичных сигналах на входах и на выходе. Таблица истинности двухвходовых элементов «И, И-НЕ, ИЛИ, ИЛИ-НЕ» представлена ниже.

Таблица 1.2

Таблица истинности двухвходовых элементов «И, И-НЕ, ИЛИ, ИЛИ-НЕ»

Вход «1»	Вход «2»	Выход «И»	Выход «И-НЕ»	Выход «ИЛИ»	Выход «ИЛИ-НЕ»
0	0	0	1	0	1
0	1	0	1	1	0
1	0	0	1	1	0
1	1	1	0	1	0



Рис. 1.6. Обозначения элементов «И, И-НЕ, ИЛИ, ИЛИ-НЕ»: зарубежные (слева) и отечественные (справа)

Отечественные и зарубежные обозначения на схемах двухвходовых элементов «И, И-НЕ, ИЛИ, ИЛИ-НЕ» показаны на рис. 1.6. Данные элементы могут быть использованы как элементы разрешения или запрещения, смешивания или совпадения. На рис. 1.7 приведены примеры формирования элементами выходных сигналов на основании требуемых временных диаграмм входных и выходных сигналов.

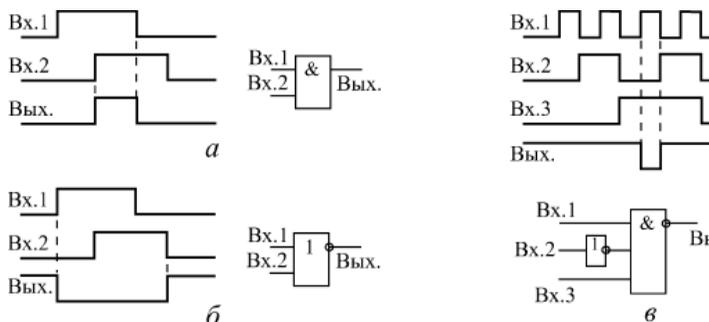


Рис. 1.7. Примеры применения элементов «И» и «ИЛИ»

В случае элемента «2И-НЕ» один из входов можно считать информационным, а другой - управляющим. В этом случае при единице на управляющем входе выходной сигнал будет равен инвертированному входному сигналу, а при нуле на управляющем входе выходной сигнал будет постоянно равен единице, то есть прохождение входного сигнала будет запрещено. Элементы «2И-НЕ» часто используют именно в качестве управляе-

мых буферов для работы на мультиплексированную или двунаправленную линию.

Точно так же в качестве элемента разрешения или запрещения могут применяться элементы «И, ИЛИ, ИЛИ-НЕ» (рис. 1.8). Разница между элементами состоит только в полярности управляющего сигнала, в инверсии (или ее отсутствии) входного сигнала, а также в уровне выходного сигнала (ноль или единица) при запрещении прохождения входного сигнала.

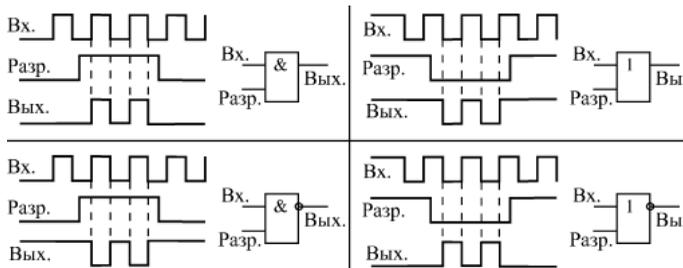


Рис. 1.8. Разрешение/запрещение прохождения сигналов на элементах «И, И-НЕ, ИЛИ, ИЛИ-НЕ»

Рассмотрим функцию смешивания двух сигналов. Если оба входных сигнала положительные и выходной сигнал положительный, то мы имеем в чистом виде функцию «ИЛИ», и требуется элемент «2ИЛИ». Однако при отрицательных входных сигналах и отрицательном выходном сигнале для такого же смешивания понадобится уже элемент «2И». А если полярность входных сигналов не совпадает с нужной полярностью выходного сигнала, то нужны уже элементы с инверсией («И-НЕ» при положительных выходных сигналах и «ИЛИ-НЕ» при отрицательных выходных сигналах). На рис. 1.9 показаны варианты смешивания на разных элементах [3].

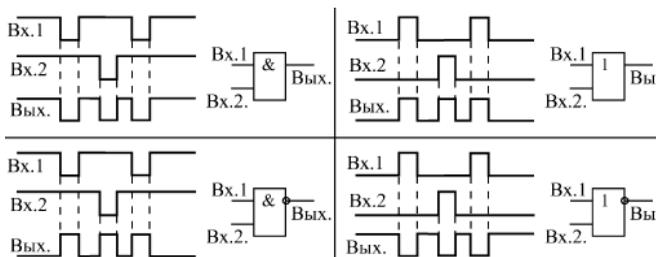


Рис. 1.9. Реализация смешивания двух сигналов

Элементы «Исключающее «ИЛИ» также можно было бы отнести к простейшим элементам, но функция, выполняемая ими, несколько сложнее, чем в случае элемента «И» или элемента «ИЛИ». Все входы элементов «Исключающее «ИЛИ»» равноправны, однако ни один из входов не может заблокировать другие входы, установив выходной сигнал в уровень единицы или нуля.

Таблица 1.3

Таблица истинности «Исключающего «ИЛИ»»

Вход «1»	Вход «2»	Выход
0	0	0
0	1	1
1	0	1
1	1	0



Рис. 1.10. Обозначения элементов «Исключающее «ИЛИ»»: зарубежные (слева) и отечественные (справа)

Под функцией «Исключающее «ИЛИ» понимается следующее: единица на выходе появляется тогда, когда только на одном входе присутствует единица. Если единиц на входах две или больше, или если на всех входах нули, то на выходе будет нуль. Таблица истинности двухвходового элемента «Исключающее «ИЛИ» приведена в табл. 1.3. Обозначения, принятые в отечественных и зарубежных схемах, показаны на рис. 1.10.

Основное применение элементов «Исключающее «ИЛИ», прямо следующее из таблицы истинности, состоит в сравнении двух входных сигналов. В случае, когда на входы приходят две единицы или два нуля (сигналы совпадают), на выходе формируется нуль. Обычно при таком применении на один вход элемента подается постоянный уровень, с которым сравнивается изменяющийся во времени сигнал, приходящий на другой вход. Но значительно чаще для сравнения сигналов и кодов применяются специальные микросхемы компараторов кодов, которые будут рассмотрены далее.

Важное применение элементов «Исключающее «ИЛИ» - это управляемый инвертор (рис. 1.11). В этом случае один из входов элемента используется в качестве управляющего, а на другой вход элемента поступает информационный сигнал. Если на управляющем входе единица, то входной сигнал инвертируется, если же нуль - не инвертируется. Чаще всего управляющий сигнал задается постоянным уровнем, определяя режим работы элемента, а информационный сигнал является импульсным. То есть элемент «Исключающее «ИЛИ» может изменять полярность входного сигнала или фронта, а может и не изменять в зависимости от управляющего сигнала.

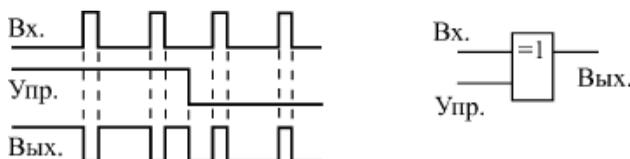


Рис. 1.11. Элемент «Исключающее «ИЛИ» как управляемый инвертор

В случае, когда имеется два сигнала одинаковой полярности (положительные или отрицательные), и при этом их одновременный приход исключается, элемент «Исключающее «ИЛИ» может быть использован для смешивания этих сигналов (рис. 1.12). При любой полярности входных сигналов выходные сигналы элемента будут положительными. При положительных входных сигналах элемент «Исключающее «ИЛИ» будет работать как элемент «2ИЛИ», а при отрицательных он будет заменять элемент «2И-НЕ». Такие замены могут быть полезны в тех случаях, когда в схеме остаются неиспользованными некоторые элементы «Исключающее «ИЛИ».

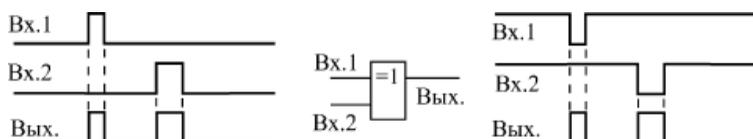


Рис. 1.12. Применение элемента «Исключающее «ИЛИ» для смешивания двух не одновременных сигналов

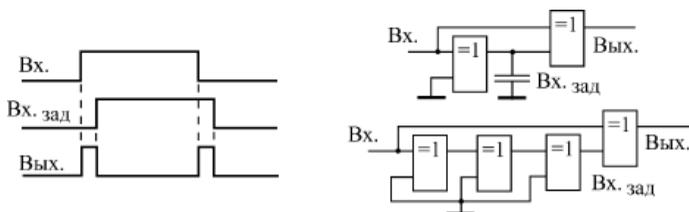


Рис. 1.13. Выделение фронтов входного сигнала с помощью элемента Исключающее ИЛИ

Еще одно важнейшее применение элемента «Исключающее «ИЛИ» - формирование коротких импульсов по любому фронту входного сигнала (рис. 1.13). В данном случае не важно, положительный фронт входного сигнала или отрицательный, на выходе все равно формируется положительный импульс. Входной сигнал задерживается с помощью конденсатора или цепочки элементов, а затем исходный сигнал и его задержанная копия

поступают на входы элемента «Исключающее «ИЛИ»». В обеих схемах в качестве элементов задержки используются также двухвходовые элементы «Исключающее «ИЛИ» в не инвертирующем включении [3].

Помимо простейших логических элементов в состав стандартных серий входит и несколько более сложные логические элементы. Они представляют собой комбинацию из простейших логических элементов. От более сложных комбинационных микросхем эти элементы отличаются именно очевидной сводимостью к простейшим элементам. Поэтому в справочниках обычно даже не приводятся таблицы истинности этих элементов.

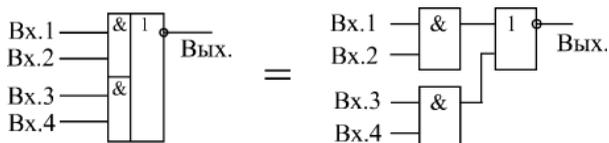


Рис. 1.14. Логический элемент «ЛР1» и его эквивалентная схема

Типичный пример сложного логического элемента – «ЛР1». В корпусе микросхемы содержится два элемента, каждый из которых представляет собой комбинацию из двух элементов «2И» и одного элемента «2ИЛИ-НЕ» (рис. 1.14). По такому же принципу строятся и другие микросхемы «ЛР». Разница между ними только в количестве элементов «И» и в количестве входов этих элементов (рис. 1.15).

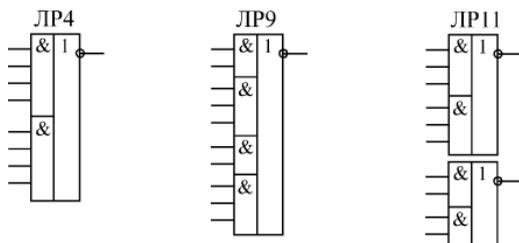


Рис. 1.15. Примеры логических элементов «ЛР»

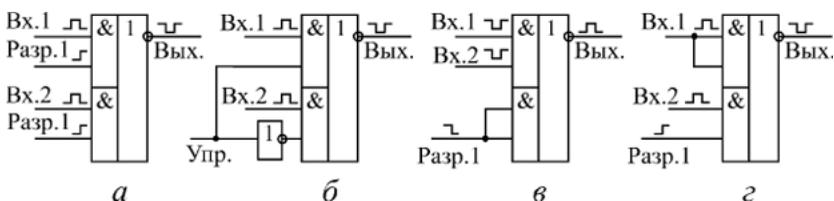


Рис. 1.16. Примеры использования элементов ЛР1

На рис. 1.16 приведено несколько примеров наиболее типичных применений микросхемы «ЛР1». Самое распространенное ее использование (рис. 1.16, а) состоит в организации двухканального мультиплексирования, то есть в переключении сигналов с двух входов на один выход. При этом один из входов каждого из элементов «2И» используется в качестве информационного, а другой - в качестве разрешающего. Вариант этого включения (рис. 1.16, б) - использование одного управляющего входа переключения каналов и дополнительного инвертора. При единице на управляющем входе работает верхний канал, при нуле - нижний. Еще один вариант использования элемента «ЛР1» (рис. 1.16, в) - смешивание двух отрицательных входных сигналов с возможностью разрешения или запрета выходного сигнала. Последний показанный на рисунке вариант (рис. 1.16, г) - смешивание двух положительных сигналов, один из которых может быть разрешен или запрещен. То есть такое объединение в одном элементе функций «И» и «ИЛИ» довольно удобно [3].

1.4. Модели описания работы цифровых устройств

Все цифровые устройства строятся из логических микросхем, каждая из которых (рис. 1.17) обязательно имеет следующие выводы:

- выводы питания: общий («земля») и напряжения питания (в большинстве случаев - +5В или +3,3В);
- выводы для входных сигналов, на которые поступают внешние цифровые сигналы;

- выводы для выходных сигналов, на которые выдаются цифровые сигналы из самой микросхемы.

Каждая микросхема преобразует тем или иным способом последовательность входных сигналов в последовательность выходных сигналов. Способ преобразования чаще всего описывается или в виде таблицы, или в виде временных диаграмм.

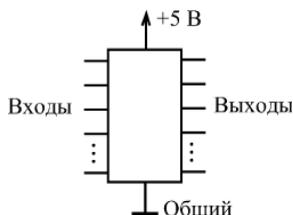


Рис. 1.17. Цифровая микросхема

Все цифровые микросхемы работают с логическими сигналами, имеющими два разрешенных уровня напряжения. Один из этих уровней называется уровнем логической единицы (или единичным уровнем), а другой - уровнем логического нуля (или нулевым уровнем). Чаще всего логическому нулю соответствует низкий уровень напряжения, а логической единице - высокий уровень. В этом случае говорят, что принята «положительная логика», если наоборот - «отрицательная логика».

Для описания работы цифровых устройств используют самые различные модели, отличающиеся друг от друга сложностью, точностью, большим или меньшим учетом тонких физических эффектов. В подавляющем большинстве случаев для разработчика цифровых схем достаточно трех моделей, трех уровней представления о работе цифровых устройств:

1. Логическая модель.
2. Модель с временными задержками.
3. Модель с учетом электрических эффектов (или электрическая модель).

Логическая модель применима для всех цифровых схем, работающих с низкой скоростью, в которых быстроедействие не

принципиально. Привлечение второй модели, учитывающей задержки срабатывания логических элементов. Ее применение необходимо для всех быстродействующих устройств и для случая одновременного изменения нескольких входных сигналов. Наконец, добавление третьей модели, учитывающей входные и выходные токи, входные и выходные сопротивления и емкости элементов, дает возможность проектирования практически всех цифровых схем. В первую очередь, эту третью модель надо применять при объединении нескольких входов и выходов, при передаче сигналов на большие расстояния и при нетрадиционном включении логических элементов.

Для иллюстрации работы перечисленных моделей рассмотрим работу самого простейшего логического элемента - инвертора. Инвертор изменяет (инвертирует) логический уровень входного сигнала на противоположный уровень выходного сигнала или изменяет полярность логического сигнала. На рис. 1.18 показано, как будет выглядеть выходной сигнал инвертора при использовании трех его моделей (трех уровней его представления). Такие графики логических сигналов называются временными диаграммами, они позволяют лучше понять работу цифровых схем.

На практике разработчик, как правило, в начале проектирования пользуется исключительно первой моделью, а затем для некоторых узлов применяет вторую или третью модель. При этом первая модель не требует вообще никаких цифровых расчетов, для нее достаточно только знание таблиц истинности или алгоритмов функционирования микросхем. Вторая модель предполагает расчет (по сути, суммирование) временных задержек элементов на пути прохождения сигналов [3].

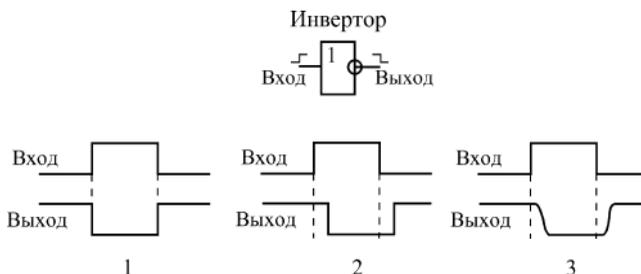


Рис. 1.18. Три уровня представления цифровых устройств

К третьему уровню относится ряд параметров, которые часто упоминаются в литературе:

Порог срабатывания - уровень входного напряжения, выше которого сигнал воспринимается как единица, а ниже - как нуль. Для наиболее распространенных ТТЛ (транзисторно-транзисторная логика) микросхем он примерно равен 1,3...1,4 В.

Помехозащищенность - характеризует величину входного сигнала помехи, накладывающегося на входной сигнал, который еще не может изменить состояние выходных сигналов.

Коэффициент разветвления - число входов, которое может быть подключено к данному выходу без нарушения работы.

Нагрузочная способность - параметр выхода, характеризующий величину выходного тока, которую может выдать в нагрузку данный выход без нарушения работы. Чаще всего нагрузочная способность прямо связана с коэффициентом разветвления.

При классической организации связей (рис. 1.19) все сигналы между устройствами передаются по своим отдельным линиям (проводам). Каждое устройство передает свои сигналы всем другим независимо от других устройств. В этом случае обычно получается большое количество линий связи, к тому же правила обмена сигналами по этим линиям (или протоколы обмена) достаточно усложнены.

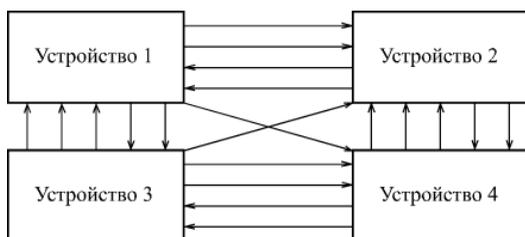


Рис. 1.19. Классическая организация связей



Рис. 1.20. Шинная организация связей

При шинной же организации связей (рис. 1.20) все сигналы между устройствами передаются по одним и тем же линиям (проводам), но в разные моменты времени (это называется временным мультиплексированием). В результате количество линий связи резко сокращается, а правила обмена сигналами существенно упрощаются. Группа линий (сигналов), используемая несколькими устройствами, как раз и называется шиной. Понятно, что объединение выходов в этом случае совершенно необходимо - ведь каждое устройство должно иметь возможность выдавать свой сигнал на общую линию. К недостаткам шинной организации относится, прежде всего, невысокая (по сравнению с классической структурой связей) скорость обмена сигналами. При простых структурах связи она может быть избыточна [3].

1.5. Основные обозначения устройств и функциональных узлов на схемах

Для изображения электронных устройств и их узлов применяется три основных типа схем:

- принципиальная схема;
- структурная схема;
- функциональная схема.

Различаются они своим назначением и степенью детализации изображения устройств.

Принципиальная схема - наиболее подробная. Она обязательно показывает все использованные в устройстве элементы и все связи между ними. Если схема строится на основе микросхем, то должны быть показаны номера выводов всех входов и выходов этих микросхем. Принципиальная схема должна позволять полностью воспроизвести устройство. Обозначения принципиальной схемы наиболее жестко стандартизованы, отклонения от стандартов не рекомендуются.

Структурная схема - наименее подробная. Она предназначена для отображения общей структуры устройства, то есть его основных блоков, узлов, частей и главных связей между ними. Из структурной схемы должно быть понятно, зачем нужно данное устройство и что оно делает в основных режимах работы, как взаимодействуют его части. Обозначения структурной схемы могут быть довольно произвольными, хотя некоторые общепринятые правила все-таки лучше выполнять.

Функциональная схема представляет собой гибрид структурной и принципиальной. Некоторые наиболее простые блоки, узлы, части устройства отображаются на ней, как на структурной схеме, а остальные - как на принципиальной схеме. Функциональная схема дает возможность понять всю логику работы устройства, все его отличия от других подобных устройств, но не позволяет без дополнительной самостоятельной работы воспроизвести это устройство. Что касается обозначений, используемых на функциональных схемах, то в части, показанной как

структура, они не стандартизованы, а в части, показанной как принципиальная схема, - стандартизованы.

Положительный сигнал (сигнал положительной полярности) - это сигнал, активный уровень которого «логическая единица». Отрицательный сигнал (сигнал отрицательной полярности) - это сигнал, активный уровень которого «логический ноль» (рис. 1.21).

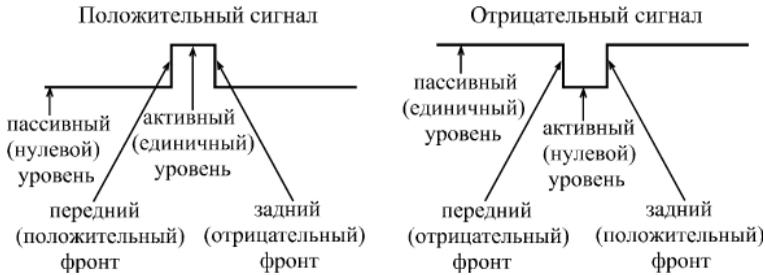


Рис. 1.21. Элементы цифрового сигнала

Активный уровень сигнала - это уровень, соответствующий приходу сигнала, то есть выполнению этим сигналом соответствующей ему функции.

Пассивный уровень сигнала - это уровень, в котором сигнал не выполняет никакой функции.

Инвертирование или инверсия сигнала - это изменение его полярности.

Инверсный выход - это выход, выдающий сигнал инверсной полярности по сравнению с входным сигналом.

Прямой выход - это выход, выдающий сигнал такой же полярности, какую имеет входной сигнал.

Положительный фронт сигнала - это переход сигнала из нуля в единицу.

Отрицательный фронт сигнала (спад) - это переход сигнала из единицы в ноль.

Передний фронт сигнала - это переход сигнала из пассивного уровня в активный.

Задний фронт сигнала - это переход сигнала из активного уровня в пассивный.

Тактовый сигнал (или строб) - управляющий сигнал, который определяет момент выполнения элементом или узлом его функции.

Шина - группа сигналов, объединенных по какому-то принципу, например, шиной называют сигналы, соответствующие всем разрядам какого-то двоичного кода.

Обозначение входов и выходов цифровых микросхем показана на рисунке ниже.



Рис. 1.22. Обозначение входов и выходов

Если у микросхемы необходимо показать неинформационные выводы, то есть выводы, не являющиеся ни логическими входами, ни логическими выходами, то такой вывод помечается косым крестом. Это могут быть, например, выводы для подключения внешних элементов (резисторов, конденсаторов) или выводы питания (рис. 1.23).

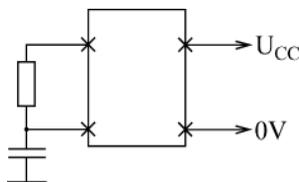


Рис. 1.23. Обозначение неинформационных выводов

В схемах также предусматриваются специальные обозначения для шин (рис. 1.24). На структурных и функциональных схемах шины обозначаются толстыми линиями или двойными стрелками, причем количество сигналов, входящих в шину, указывается рядом с косой чертой, пересекающей шину. На принципиальных схемах шина тоже обозначается толстой линией, а входящие в шину и выходящие из шины сигналы изображаются в виде перпендикулярных к шине тонких линий с указанием их номера или названия. При передаче по шине двоичного кода нумерация начинается с младшего разряда кода.

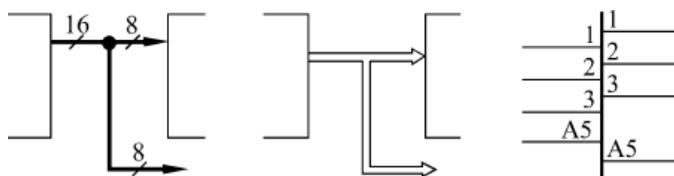


Рис. 1.24. Обозначение шин

При изображении микросхем используются сокращенные названия входных и выходных сигналов, отражающие их функцию. Эти названия располагаются на рисунке рядом с соответствующим выводом. Также на изображении микросхем указывается выполняемая ими функция. Изображение микросхемы иногда делят на три вертикальные поля. Левое поле относится к входным сигналам, правое - к выходным сигналам. В центральном поле помещается название микросхемы и символы ее особенностей. Неинформационные выводы могут указываться как на левом, так и на правом поле; иногда их показывают на верхней или нижней стороне прямоугольника, изображающего микросхему [3].

1.6. Серии цифровых микросхем

В настоящее время выпускается огромное количество разнообразных цифровых микросхем: от простейших логиче-

ских элементов до сложнейших процессоров, микроконтроллеров и специализированных БИС. Производством цифровых микросхем занимается множество фирм - как у нас в стране, так и за рубежом. Поэтому даже классификация этих микросхем представляет собой довольно трудную задачу.

Однако в качестве базиса в цифровой схемотехнике принято рассматривать классический набор микросхем малой и средней степени интеграции, в основе которого лежат ТТЛ серии семейства 74, выпускаемые уже несколько десятилетий рядом фирм.



Рис. 1.25. Система обозначений фирмы «Texas Instruments»

В качестве примера рассмотрим систему обозначений фирмы «Texas Instruments» (рис. 1.25). Полное обозначение состоит из шести элементов:

Идентификатор фирмы SN (для серий AC и ACT отсутствует).

Тип семейства:

74 - коммерческие микросхемы;

54 - микросхемы военного назначения.

Код серии (до трех символов):

Отсутствует - стандартная ТТЛ - серия.

LS (Low Power Schottky) - маломощная серия ТТЛШ;

S (Schottky) - серия ТТЛШ;

ALS (Advanced Schottky) – улучшенная серия ТТЛШ;

F (FAST) - быстрая серия;

HC (High Speed CMOS) - высокоскоростная КМОП - серия;

HCT (High Speed CMOS with TTL inputs) - серия HC, совместимая по входу с TTL;

AC (Advanced CMOS) - улучшенная серия КМОП;

ACT (Advanced CMOS with TTL inputs) - серия AC, совместимая по входу с TTL;

BCT (BiCMOS Technology) - серия с БиКМОП - технологией;

ABT (Advanced BiCMOS Technology) - улучшенная серия с БиКМОП - технологией.

LVT (Low Voltage Technology) - серия с низким напряжением питания.

Код типа корпуса:

N - пластмассовый корпус DIL (DIP);

J - керамический DIL (DIC);

T - плоский металлический.



Рис. 1.26. Обозначения отечественных микросхем

Отечественная система обозначений микросхем отличается от рассмотренной довольно существенно (рис. 1.26). Основные элементы обозначения следующие:

Буква К обозначает микросхемы широкого применения, для микросхем военного назначения буква отсутствует.

Тип корпуса микросхемы:

P - пластмассовый корпус;

M - керамический;

B - бескорпусная микросхема.

Номер серии микросхем (от трех до четырех цифр).

Функция микросхемы (две буквы).

Номер микросхемы (от одной до трех цифр).

Главное достоинство отечественной системы обозначений состоит в том, что по обозначению микросхемы можно легко понять ее функцию. Зато в системе обозначений «Texas Instruments» виден тип серии с его особенностями [3].

2. ОСНОВНЫЕ ФУНКЦИОНАЛЬНЫЕ УЗЛЫ ИНТЕГРАЛЬНЫХ СХЕМ

2.1. Функциональные узлы комбинационного и последовательностного типа

Функциональные узлы и цифровые устройства делятся на комбинационные и последовательностные. В дальнейшем комбинационные узлы будем обозначать через КЦ (комбинационные цепи), а последовательностные через АП (автоматы с памятью). Различия между КЦ и АП имеют фундаментальный характер. Выходные величины КЦ зависят только от текущего значения входных величин (аргументов) в отличие от АП.

Комбинационные микросхемы выполняют более сложные функции, чем простые логические элементы. Их входы объединены в функциональные группы и не являются полностью взаимозаменяемыми.

Объединяет комбинационные микросхемы с логическими элементами то, что они не имеют внутренней памяти. То есть уровни их выходных сигналов всегда однозначно определяются текущими уровнями входных сигналов и никак не связаны с предыдущими значениями входных сигналов. Любое изменение входных сигналов обязательно изменяет состояние выходных сигналов. Именно поэтому логические элементы иногда также называют комбинационными микросхемами, в отличие последовательностных (АП) микросхем, которые имеют внутреннюю память и управляются не уровнями входных сигналов, а их последовательностями.

Все комбинационные микросхемы внутри построены из простейших логических элементов, и эта их внутренняя структура часто приводится в справочниках. Внутренняя структура важна для разработчиков микросхем, а также в тех редчайших случаях, когда надо построить новую комбинационную микросхему из микросхем простых логических элементов.

Состав набора комбинационных микросхем, входящих в стандартные серии, был определен исходя из наиболее часто встречающихся задач. Требуемые для этого функции реализова-

ны в комбинационных микросхемах наиболее оптимально, с минимальными задержками и минимальным потреблением мощности [3].

2.2. Дешифраторы и шифраторы

Дешифратор преобразует входной двоичный код в номер выходного сигнала (дешифрирует код), а шифратор преобразует номер входного сигнала в выходной двоичный код (шифрует номер входного сигнала). Количество выходных сигналов дешифратора и входных сигналов шифратора равно количеству возможных состояний двоичного кода, то есть 2^n , где n - разрядность двоичного кода (рис. 2.1). Микросхемы дешифраторов обозначаются на схемах буквами DC (от английского Decoder), а микросхемы шифраторов — CD (от английского Coder).

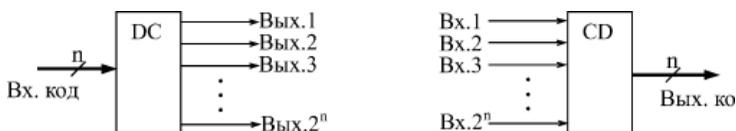


Рис. 2.1. Функции дешифратора (слева) и шифратора (справа)

На выходе дешифратора всегда присутствует только один сигнал, причем номер этого сигнала однозначно определяется входным кодом. Выходной код шифратора однозначно определяется номером входного сигнала.

Двоичный дешифратор, имеющий n входов, должен иметь 2^n выходов, соответствующих числу разных комбинаций в n -разрядном двоичном коде.

В зависимости от входного двоичного кода на выходе дешифратора возбуждается одна и только одна из выходных цепей. Если часть входных наборов не используется, то дешифратор называют неполным, и у него число выходов меньше 2^n .

В условном обозначении дешифраторов проставляются буквы DC (от англ. Decoder). Входы дешифратора принято обозначать их двоичными весами. Кроме информационных входов

дешифратор обычно имеет один или более входов разрешения работы обозначаемых как EN (Enable).

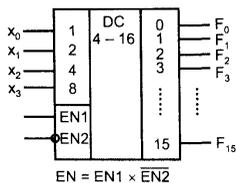


Рис. 2.2. Условное обозначение дешифратора

Условное графическое обозначение полного двоичного дешифратора показано на рис. 2.2. Часто дешифратор имеет инверсные выходы. В этом случае только один выход имеет нулевое значение, а все остальные - единичное. При запрещении работы дешифратора на всех его выходах будет присутствовать логическая единица.

Наиболее типичное применение дешифраторов состоит именно в дешифрировании входных кодов, при этом входы С используются как стробирующие, управляющие сигналы. Номер активного (то есть нулевого) выходного сигнала показывает, какой входной код поступил. Если нужно дешифровать код с большим числом разрядов, то можно объединить несколько микросхем дешифраторов (рис. 2.3).

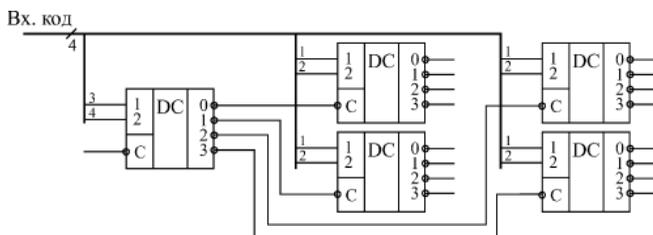


Рис. 2.3. Увеличение количества разрядов дешифратора

Шифраторы используются гораздо реже, чем дешифраторы. Это связано с более специфической областью их примене-

ния. Значительно меньше и выбор микросхем шифраторов в стандартных сериях.

Двоичные шифраторы выполняют операцию, обратную по отношению к операции дешифратора: они преобразуют код "1 из N" в двоичный. При возбуждении одного из входов шифратора на его выходе формируется двоичный код номера возбужденной входной линии. Полный двоичный шифратор имеет 2^n входов и n выходов.

На рис. 2.4 показаны для примера две микросхемы шифраторов ИВ1 и ИВ3. Первая имеет 8 входов и 3 выхода (шифратор 8–3), а вторая — 9 входов и 4 выхода (шифратор 9–4). Все входы шифраторов - инверсные (активные входные сигналы - нулевые). Все выходы тоже инверсные, то есть формируется инверсный код [1,3].

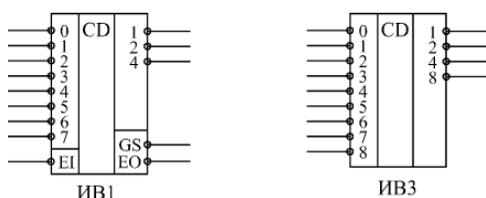


Рис. 2.4. Микросхемы шифраторов

2.3. Мультиплексоры и демультиплексоры

Мультиплексоры предназначены для поочередной передачи на один выход одного из нескольких входных сигналов, то есть для их мультиплексирования. Количество мультиплексируемых входов называется количеством каналов мультиплексора, а количество выходов называется числом разрядов мультиплексора.

Управление работой мультиплексора (выбор номера канала) осуществляется с помощью входного кода адреса. Например, для 4-канального мультиплексора необходим 2-разрядный управляющий (адресный) код, а для 16-канального 4-разрядный код. Разряды кода обозначаются 1, 2, 4, 8 или A0, A1, A2, A3.

Некоторые микросхемы мультиплексоров имеют вход разрешения/запрета С (другое обозначение - S), который при запрете устанавливает прямой выход в нулевой уровень.

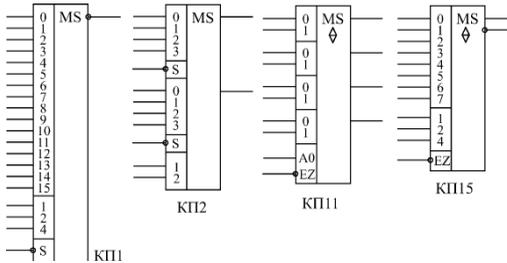


Рис. 2.5. Примеры микросхем мультиплексоров

Мультиплексоры осуществляют подключение одного из входных каналов к выходному под воздействием соответствующего управляющего (адресующего) слова. Разрядности каналов могут быть различными, мультиплексоры для коммутации многоразрядных слов составляются из одnorазрядных.

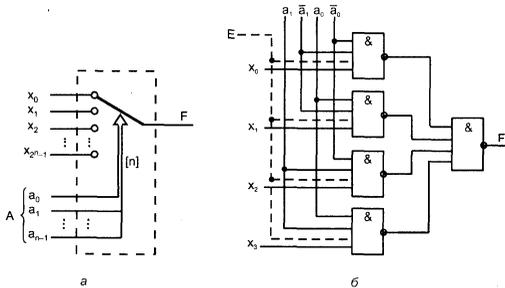


Рис. 2.6. Упрощенное представление мультиплексора многопозиционным ключом (а) и реализация мультиплексора на элементах «И-НЕ» (б)

Входы мультиплексора делятся на две группы: информационные и адресующие. Работу мультиплексора можно упрощенно представить с помощью многопозиционного ключа. Для

одноразрядного мультиплексора это представлено на рис. 2.6, а. Адресующий код А задает переключателю определенное положение, соединяя с выходом F один из информационных входов x_j . При нулевом адресующем коде переключатель занимает верхнее положение x_0 , с увеличением кода на единицу переходит в соседнее положение x_1 и т. д.

Схемотехнически мультиплексор реализует электронную версию показанного переключателя, имея, в отличие от него, только одностороннюю передачу данных. На рис. 2.6, б показан мультиплексор с четырьмя информационными входами, двумя адресными входами и входом разрешения работы. При отсутствии разрешения работы ($E = 0$) выход F становится нулевым независимо от информационных и адресных сигналов. В стандартных сериях размерность мультиплексоров не более «16 - 1».

Микросхемы мультиплексоров можно объединять для увеличения количества каналов. Например, два 8-канальных мультиплексора объединяются в 16-канальный с помощью инвертора на входах разрешения и элемента «2И-НЕ» для смешивания выходных сигналов (рис. 2.7). Старший разряд кода будет при этом выбирать один из двух мультиплексоров. Точно так же из двух 16-канальных мультиплексоров можно сделать 32-канальный. Если нужно большее число каналов, то необходимо вместо инвертора включать дешифратор, на который подаются старшие разряды кода. Выходные сигналы дешифратора будут выбирать один из мультиплексоров [3].

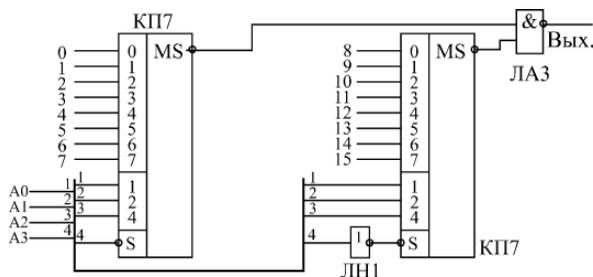


Рис. 2.7. Объединение мультиплексоров для увеличения количества каналов

Демультимплексоры выполняют операцию, обратную операции мультиплексов - передают данные из одного входного канала в один из нескольких каналов - приемников. Многоуровневые демультимплексоры состояются из нескольких одноуровневых. Условное обозначение демультимплексов на примере размерности «1 - 4» показано на рис. 2.8 [3].

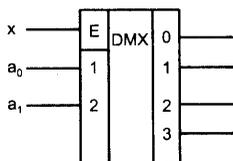


Рис. 2.8. Условное обозначение демультимплексора

2.4. Компараторы кодов

Микросхемы компараторов кодов применяются для сравнения двух входных кодов и выдачи на выходы сигналов о результатах этого сравнения (о равенстве или неравенстве кодов). На схемах компараторы кодов обозначаются двумя символами равенства: «= =». Код типа микросхемы компаратора кода в отечественных сериях - СП.

Примером такой микросхемы может служить СП1 - четырех разрядный компаратор кодов, сравнивающий величины кодов и выдающий информацию о том, какой код больше, или о равенстве кодов (рис. 2.10).

Компараторы (устройства сравнения) определяют отношения между двумя словами. Основными отношениями, через которые можно выразить остальные, можно считать два - "равно" и "больше".

Эти отношения используются как логические условия в микропрограммах, в устройствах контроля и диагностики ЭВМ и т. д.

В сериях цифровых элементов обычно имеются компараторы с тремя выходами: "равно", "больше" и "меньше" (рис. 2.9).

Для краткости записей в индексе выходных функций указывается только слово А [4].

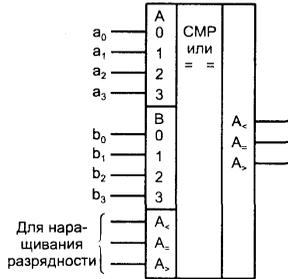


Рис. 2.9. Условное обозначение компаратора с тремя выходами

Если используется одиночная микросхема, то для ее правильной работы достаточно подать единицу на вход $A = B$, а состояния входов $A < B$ и $A > B$ не важны, на них можно подать как нуль, так и единицу. Назначение выходов понятно из их названия, а полярность выходных сигналов положительная. Если микросхемы компараторов кодов каскадируются (объединяются) для увеличения числа разрядов сравниваемых кодов, то надо выходные сигналы микросхемы, обрабатывающей младшие разряды кода, подать на одноименные входы микросхемы, обрабатывающей старшие разряды кода (рис. 2.10) [4].

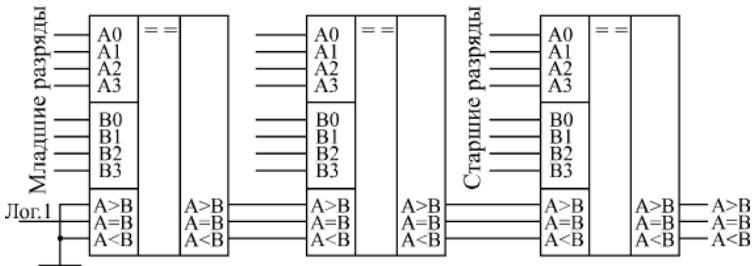


Рис. 2.10. Каскадирование компараторов кодов

2.5. Триггерные устройства. Классификация. Основные сведения

Триггеры - элементарные автоматы (функциональные узлы последовательностного типа), содержащие собственно элемент памяти (фиксатор) и схему управления. Фиксатор строится на двух инверторах, связанных друг с другом «накрест», так что выход одного соединен с входом другого. Такое соединение дает цепь с двумя устойчивыми состояниями (рис. 2.11). Действительно, если на выходе инвертора 1 имеется логический нуль, то он обеспечивает на выходе инвертора 2 логическую единицу, благодаря которой сам и существует. То же согласование сигналов имеет место и для второго состояния, когда инвертор 1 находится в единице, а инвертор 2 - в нуле. Любое из двух состояний может существовать неограниченно долго.

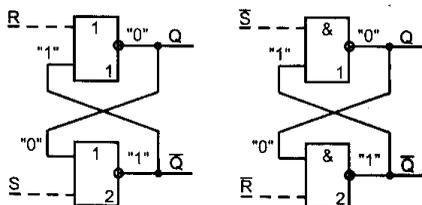


Рис. 2.11. Схемы фиксаторов с входами управления на элементах «ИЛИ-НЕ» и «И-НЕ»

Чтобы управлять фиксатором, нужно иметь в логических элементах дополнительные входы, превращающие инверторы в элементы «И-НЕ» либо «ИЛИ-НЕ». На входы управления поступают внешние установочные сигналы.

Установочные сигналы показаны на рис. 2.12 штриховыми линиями. Буквой R латинского алфавита (от Reset) обозначен сигнал установки триггера в нуль (сигнал сброса), а буквой S (от Set) - сигнал установки в состояние логической единицы (сигнал установки). Состояние триггера считывается по значению прямого выхода, обозначаемого как Q. Чаще всего триггер имеет и

второй выход с инверсным сигналом Q . Для фиксатора на элементах «ИЛИ-НЕ» установочным сигналом является единичный, поскольку только он приводит логический элемент и нулевое состояние независимо от сигналов на других входах элемента. Для фиксатора на элементах «И-НЕ» установочным сигналом является нулевой, как обладающий тем же свойством однозначно задавать состояние элемента независимо от состояний других входов.

Практически все серии цифровых ИС содержат готовые триггеры, и поэтому задача проектировщика - правильное использование имеющихся триггеров. Отсюда большое значение приобретает классификация триггеров, изучение их параметров и особенностей функционирования.

Классификация триггеров проводится по признакам логического функционирования и способу записи информации (рис. 2.12).

По логическому функционированию различают триггеры типов RS, D, T, JK и др. Кроме того, используются комбинированные триггеры, в которых совмещаются одновременно несколько типов, и триггеры со сложной входной логикой (группами входов, связанных между собой логическими зависимостями).

Триггер типа RS имеет два входа - установки в единицу (S) и установки в нуль (R).

Одновременная подача сигналов установки S и сброса R не допускается (эта комбинация сигналов называется запрещенной).

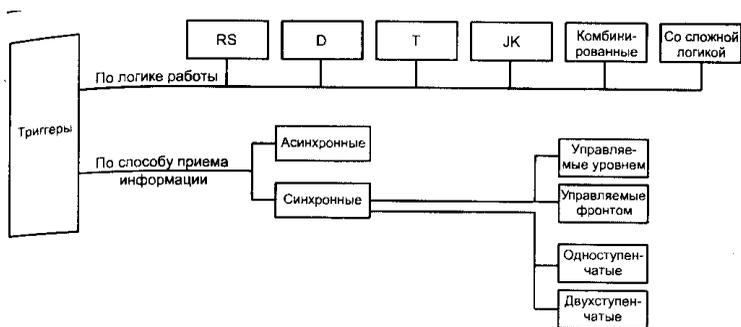


Рис. 2.12. Классификация триггеров

Триггер типа D (триггеры задержки) имеет один вход. Его состояние повторяет входной сигнал, но с задержкой, определяемой тактовым сигналом.

Триггер типа T изменяет свое состояние каждый раз при поступлении входного сигнала. Такой триггер имеет один вход и называется триггером со счетным входом или счетным триггером.

Триггер типа JK универсален, имеет входы установки (J) и сброса (K), подобные входам триггера RS. В отличие от последнего, допускает ситуацию с одновременной подачей сигналов на оба эти входа ($J = K = 1$). В этом режиме работает как счетный триггер относительно третьего (тактового) входа. По способу записи информации различают асинхронные (нетактируемые) и синхронные (тактируемые) триггеры. В нетактируемых переход в новое состояние вызывается непосредственно изменениями входных информационных сигналов. В тактируемых триггерах, имеющих специальный вход, переход происходит только при подаче на этот вход тактовых сигналов. Тактовые алы называют также синхронизирующими, исполнительными, командами и т. д. Обозначаются они буквой С (от слова Clock) [4].

По способу восприятия тактовых сигналов триггеры делятся на управляемые нем и управляемые фронтом. Управление уровнем означает, что при одном состоянии тактового сигнала

триггер воспринимает входные сигналы и реагирует на них, а при другом не воспринимает и остается в неизменном состоянии. При управлении фронтом разрешение на переключение дается только в момент перепада тактового сигнала (на его фронте или спаде). В остальное время независимо от уровня тактового сигнала триггер не воспринимает входные сигналы и остается в неизменном состоянии. Триггеры, управляемые фронтом, называют также триггерами с динамическим управлением.

Динамический вход может быть прямым или инверсным. Прямое динамическое управление означает разрешение на переключение при изменении тактового сигнала с нулевого значения на единичное, инверсное - при изменении тактового сигнала с единичного значения на нулевое.

По характеру процесса переключения триггеры делятся на одноступенчатые и двухступенчатые. В одноступенчатом триггере переключение в новое состояние происходит сразу, в двухступенчатом - по этапам. Двухступенчатые триггеры состоят из входной и выходной ступеней. Переход в новое состояние происходит в обеих ступенях поочередно. Один из уровней тактового сигнала разрешает прием информации во входную ступень при неизменном состоянии выходной ступени. Другой уровень тактового сигнала разрешает передачу нового состояния из входной ступени в выходную [4].



Рис. 2.13. Временные диаграммы, поясняющие работу синхронных триггеров, и условные обозначения тактирующих входов

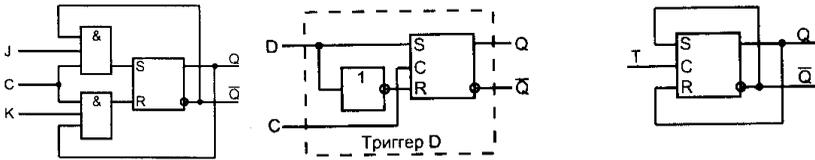


Рис. 2.14. Схемы D, T и JK-триггеров

Схема связей, отражающих структуры D, T и JK-триггеров, показаны на рис. 2.14.

3. ЗАПОМИНАЮЩИЕ УСТРОЙСТВА

3.1. Основные сведения. Система параметров.

Классификация

Запоминающие устройства (ЗУ) служат для хранения информации и обмена ею с другими ЦУ. Микросхемы памяти в общем объеме выпуска ИС занимают около 40% и играют важнейшую роль во многих системах различного назначения. Микросхемы и системы памяти постоянно совершенствуются как в области схемотехнологии, так и в области развития новых архитектур. В настоящее время созданы и используются десятки различных типов ЗУ.

Важнейшие параметры ЗУ находятся между собой в противоречии. Так, например, большая информационная емкость не сочетается с высоким быстродействием, а быстродействие в свою очередь не сочетается с низкой стоимостью. Поэтому системам памяти свойственна многоступенчатая иерархическая структура, и в зависимости от роли того или иного ЗУ его реализация может быть существенно различной.

В развитой иерархии памяти ЭВМ можно выделить следующие уровни:

- регистровые ЗУ, находящиеся в составе процессора, благодаря которым уменьшается число обращений к другим уровням памяти, реализованным вне процессора и требующим большего времени для операций обмена информацией;

- кэш-память, служащая для хранения копий информации, используемой в текущих операциях обмена. Работа процессора с кэш-памятью высокого быстродействия повышает производительность ЭВМ;

- основная память (оперативная, постоянная, полупостоянная), работающая в режиме обмена с процессором и по возможности согласованная с ним по быстродействию. Исполняемый в текущий момент фрагмент программы обязательно находится в основной памяти;

- специализированные виды памяти, характерные для некоторых специфических архитектур (многопортовые, ассоциативные, видеопамять и др.);

- внешняя память, хранящая большие объемы информации. Эта память обычно реализуется на основе устройств с подвижным носителем информации (магнитные и оптические диски, магнитные ленты и др.). В настоящем пособии устройства внешней памяти не рассматриваются [4].

3.2. Параметры запоминающих устройств

Информационная емкость - максимально возможный объем хранимой информации. Выражается в битах или словах (в частности, в байтах). Бит хранится запоминающим элементом (ЗЭ), а слово - запоминающей ячейкой (ЗЯ), т. е. группой ЗЭ, к которым возможно лишь одновременное обращение.

Организация ЗУ - произведение числа хранимых слов на их разрядность. Видно, что это произведение выражает информационную емкость ЗУ, однако при одной и той же информационной емкости организация ЗУ может быть различной, так что организация является самостоятельным важным параметром, выражаемым парой чисел.

Быстродействие ЗУ оценивают временами считывания, записи, длительностями циклов чтения/записи и другими параметрами. Время считывания - интервал между моментами появления сигнала чтения и слова на выходе ЗУ. Время записи - интервал после появления сигнала записи, достаточный для установления ЗЯ в состояние, задаваемое входным словом. Минимально допустимый интервал между последовательными повторными операциями чтения или записи образует соответствующий цикл. Длительности циклов могут превышать времена чтения или записи, т. к. после этих операций до начала следующей может потребоваться время для восстановления необходимого начального состояния ЗУ.

Времена чтения, записи и длительности циклов - традиционные параметры, достаточные для оценки быстродействия

простых структур ЗУ. Для многих современных ЗУ они должны быть дополнены новыми. Причиной является более сложный характер доступа к хранимым данным, когда обращение к первому слову некоторой группы слов (страницы, пакета) требует большего времени, чем обращение к последующим. Для таких режимов вводят параметры времени доступа при первом обращении (Latency) и темпа передач для последующих слов пакета. Темп передач в свою очередь оценивается двумя значениями - предельным (внутри пакета) и усредненным (с учетом Latency). С уменьшением пакета усредненный темп снижается, все более отличаясь от предельного.

Применительно к ЗУ используется также параметр, называемый полосой пропускания или производительностью, и определяемый как произведение числа считываемых (или записываемых) в секунду слов на их разрядность. Например, ЗУ с темпом передачи слов 50 МГц при их разрядности 8 бит имеет полосу пропускания (производительность) 400 Мбит/с.

Указанные параметры не исчерпывают всего множества сведений о быстродействии разных типов ЗУ, имеющих свою специфику. Для некоторых структур в число параметров, характеризующих быстродействие ЗУ, входят еще несколько показателей, о которых говорится далее в этой главе при описании конкретных разновидностей микросхем памяти.

Перечисленные динамические параметры являются эксплуатационными (измеряемыми). Кроме них существует ряд режимных параметров, обеспечение которых необходимо для нормального функционирования ЗУ, поскольку оно имеет несколько сигналов управления, сигналы адресации данных и самих данных и для них должно быть обеспечено определенное взаимное положение во времени. Для этих сигналов задаются длительности и ограничения по взаимному положению во времени.

Важным для микросхем памяти является свойство энергонезависимости, т. е. способность ЗУ сохранять данные при отключении напряжения питания. Энергонезависимость может быть естественной, т. е. присущей самим ЗЭ, или искусственной,

достигаемой введением резервных источников питания, автоматически подключаемых к накопителю ЗУ при снятии основного питания или же дополнением схемы ЗУ специальными вспомогательными энергонезависимыми элементами памяти.

Кроме отмеченных параметров ЗУ характеризуются и другими (уровни напряжений, токи, емкости выводов, температурный диапазон и т. д.), которые в силу своей традиционности не требуют специального рассмотрения [4].

3.3. Входные и выходные сигналы запоминающих устройств

Один из возможных наборов сигналов ЗУ (рис. 3.1, а) включает следующие сигналы (сигналы показаны для операции чтения, ЗУ обозначено буквой М от слова Memory):

A - адрес, разрядность которого n определяется числом ячеек ЗУ, т. е. максимально возможным числом хранимых в ЗУ слов. Число ячеек ЗУ выражается целой степенью двойки. Адрес является номером ячейки, к которой идет обращение. Очевидно, что разрядность адреса n и число ячеек N связаны соотношениями $n = \log_2 N$ и $N = 2^n$;

CS (Chip Select) или CE (Chip Enable), который разрешает или запрещает работу данной микросхемы;

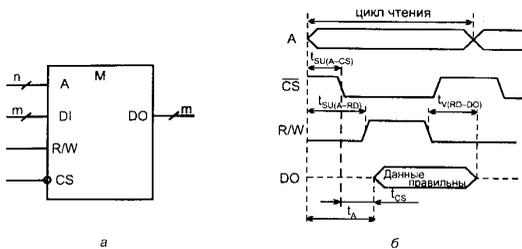


Рис. 3.1. Типичные сигналы простейшего ЗУ (а) и их временные диаграммы (б)

R/W - (Read/Write) задает выполняемую операцию (при единичном значении - чтение, при нулевом - запись);

DI и DO (Data Input) и (Data Output) - шины входных и выходных данных, разрядность которых m определяется организацией ЗУ (разрядностью его ячеек). В некоторых ЗУ эти линии объединены. При этом единая двунаправленная шина передачи данных обозначается как DIO (Data Input/Output).

Для ЗУ характерна такая последовательность сигналов. Прежде всего, подается адрес, чтобы последующие операции не коснулись какой-либо ячейки, кроме выбранной. Затем разрешается работа микросхемы сигналом CS (CE) и подается строб чтения/записи R/W (взаимное положение сигналов CS и R/W для разных ЗУ может быть различным). Если задана, например, операция чтения, то после подачи перечисленных сигналов ЗУ готовит данные для чтения, что требует определенного времени. Задний фронт сигнала R/W, положение которого во времени должно обеспечивать установление правильных данных на выходе ЗУ, считывает данные.

Пример временной диаграммы для рассмотренного набора сигналов ЗУ и операции чтения приведен на рис. 3.1, б.

Индексом A (от слова Access) обозначаются времена доступа - интервалы времени от появления того или иного управляющего сигнала до появления информационного сигнала на выходе. Время доступа относительно сигнала адреса обозначается, если следовать правилу, как $t_{A(A)}$, но часто просто как t_A . Аналогично этому, время доступа относительно сигнала CS, т. е. $t_{A(CS)}$ часто обозначается просто как t_A . Время t_A называют также временем выборки, а время t_{CS} - временем выбора [4].

3.4. Классификация современных запоминающих устройств

Для классификации ЗУ (рис. 3.2) важнейшим признаком является способ доступа к данным.

При адресном доступе код на адресных входах указывает ячейку, с которой ведется обмен. Все ячейки адресной памяти в момент обращения равнодоступны. Эти ЗУ наиболее разработаны, и другие виды памяти часто строят на основе адресной с со-

ответствующими модификациями. Адресные ЗУ делятся на RAM (Random Access Memory) и ROM (Read-Only Memory). Русские синонимы термина RAM: ОЗУ (оперативные ЗУ) или ЗУПВ (ЗУ с произвольной выборкой). Оперативные ЗУ хранят данные, участвующие в обмене при исполнении текущей программы, которые могут быть изменены в произвольный момент времени. Современные ОЗУ, как правило, не обладают энергонезависимостью (этим свойством обладают новые перспективные варианты ОЗУ, которые, возможно, вскоре начнут заменять существующие).

В ROM (русский эквивалент - ПЗУ, т. е. постоянные ЗУ) содержимое либо вообще не изменяется, либо изменяется, но редко и в специальном режиме. Для рабочего режима это "память только для чтения".

Постоянная память типа Mask ROM, обозначенная как ROM(M), программируется при изготовлении методами интегральной технологии с помощью масок. На русском языке ее можно назвать памятью типа ПЗУМ (ПЗУ масочные). Для потребителя это в полном смысле слова постоянная память, т. к. изменить ее содержимое он не может.

В следующих четырех разновидностях ROM в обозначениях присутствует буква P (от Programmable). Это программируемая пользователем память (в русской терминологии ППЗУ). В память типов PROM и EPROM-OTP содержимое записывается однократно (OTP означает One Time Programmable). В ЗУ типов EPROM, EEPROM и FLASH содержимое может быть изменено путем стирания старой информации и записи новой. В памяти EPROM (Erasable Programmable ROM) стирание выполняется облучением кристалла ультрафиолетовыми лучами, ее русское название РПЗУ-УФ (репрограммируемое ПЗУ с УФ-стиранием).

В EEPROM или, иначе, E²PROM (Electrically Erasable Programmable ROM) стирание производится электрическими сигналами, ее русское название РПЗУ-ЭС (репрограммируемое ПЗУ с электрическим стиранием) или ЭСПЗУ (электрически стираемое программируемое ПЗУ). Запись данных в EPROM, EEPROM и FLASH производится электрическими сигналами.

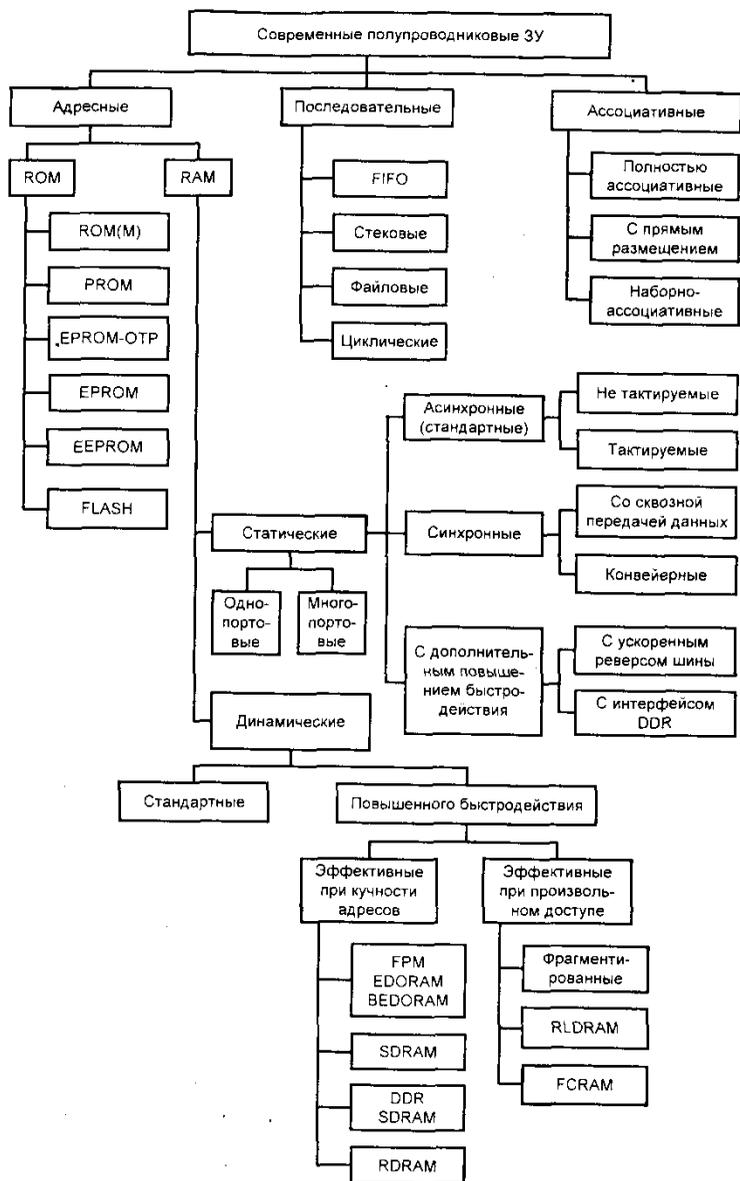


Рис. 3.2. Классификация современных полупроводниковых ЗУ

Программирование PROM, EPROM и EEPROM производится в обычных лабораторных условиях. Для ЗУ типов PROM и EPROM это делается с помощью программаторов. Для EEPROM возможно также использование специальных режимов для программирования без изъятия микросхемы из устройства, в котором она используется. Запоминающие элементы памяти типа FLASH принципиально подобны применяемым в EPROM и EEPROM, но эта память имеет структурные и технологические особенности, позволяющие выделить ее в отдельный вид.

RAM делятся на статические и динамические. В статических RAM запоминающими элементами являются триггеры, сохраняющие свое состояние, пока схема находится под питанием и нет новой записи данных. В динамических RAM данные хранятся в виде зарядов конденсаторов, образуемых элементами МОП - структур. Саморазряд конденсаторов ведет к разрушению данных, поэтому они должны периодически (каждые несколько миллисекунд) регенерироваться, что усложняет эксплуатацию ЗУ. В то же время плотность упаковки элементов динамической памяти в несколько раз превышает плотность упаковки, достижимую в статических RAM, поэтому динамические ЗУ имеют намного более высокую информационную емкость и в несколько раз дешевле более быстродействующих статических.

Разработаны также ЗУ с динамическими запоминающими элементами, имеющие внутреннюю встроенную систему регенерации, у которых внешнее поведение становится аналогичным поведению статических ЗУ. Такие ЗУ иногда называют квазистатическими (одна из фирм называет их даже статическими).

Статические ОЗУ в английской и международной терминологии называются SRAM (Static RAM), а динамические - DRAM (Dynamic RAM).

Статические ОЗУ разделены в классификации на асинхронные и синхронные. Асинхронные ОЗУ названы также стандартными, т. к. до недавнего времени они были практически единственными представителями статических микросхем памя-

ти и наиболее привычны для потребителя. В асинхронных ЗУ после произвольного по времени обращения к памяти до выдачи данных проходит определенное время, которое является параметром самой памяти, не связанным с параметрами системы синхронизации процессора. Ввиду отсутствия увязки моментов обращения к памяти и моментов выработки ею готовых данных с синхросигналами процессора могут возникать дополнительные задержки обмена данными между ЗУ и процессором.

Асинхронные статические ОЗУ можно разделить на неактивируемые и активируемые. В неактивируемых сигналы управления могут задаваться как импульсами, так и уровнями. В активируемых ЗУ некоторые сигналы обязательно должны быть импульсными (например, сигнал разрешения работы CS в каждом цикле должен переходить из пассивного состояния в активное). В синхронных ОЗУ длительности этапов работы памяти жестко связаны с синхросигналами системы, и это позволяет исключить неоправданные потери времени при обмене данными между памятью и процессором, а также организовать конвейерную обработку данных. Таким образом, синхронность памяти является средством повышения ее быстродействия. Это важный способ повышения быстродействия, применяемый как в статических, так и в динамических микросхемах памяти (в динамических ОЗУ синхронные варианты появились раньше, чем в статических). Применение синхронных ОЗУ не является единственным способом повышения их быстродействия. Среди других методов повышения быстродействия статических ОЗУ можно назвать ускорение реверса шины при переходе от передачи данных в одном направлении к другому и использование интерфейса DDR.

Статические ОЗУ выполняются как однопортовые (обычные) и многопортовые. Многопортовые ЗУ специализированы для определенных применений. В них возможны одновременные обращения более чем к одной ячейке, например, в двухпортовых ЗУ возможно считывание информации из одной ячейки и одновременная запись в другую. Подобные режимы полезны при разделении памяти между двумя или более абонентами.

Динамические ЗУ характеризуются наибольшей информационной емкостью и невысокой стоимостью, поэтому именно они используются как основная память ЭВМ. Базовая структура динамических ЗУ названа стандартной. Поскольку желательно получить от основной памяти ЭВМ максимально возможное быстродействие, разработаны многочисленные способы его повышения.

Статические ЗУ в 4 - 5 раз дороже динамических и приблизительно во столько же раз меньше по максимально достижимой информационной емкости. Их достоинством является высокое быстродействие, а типичной областью использования - схемы кэш-памяти, буферы FIFO и LIFO, память данных небольшой емкости для микроконтроллеров, быстродействующих коммуникационных устройств и т. п.

В ЗУ с последовательным доступом записываемые данные образуют некоторую очередь. Считывание происходит из очереди слово за словом либо в порядке записи, либо в обратном порядке. Моделью такого ЗУ является последовательная цепочка запоминающих элементов, в которой данные передаются между соседними элементами.

Прямой порядок считывания имеет место в буферах FIFO с дисциплиной «первый пришел - первый вышел» (First In - First Out), а также в файловых и циклических ЗУ.

Разница между памятью FIFO и файловым ЗУ состоит в том, что в FIFO запись в пустой буфер сразу же становится доступной для чтения, т. е. слово поступает в конец цепочки (модели ЗУ). В файловых ЗУ данные поступают в начало цепочки и появляются на выходе после некоторого числа обращений, равного числу элементов в цепочке. При независимости операций считывания и записи фактическое расположение данных в ЗУ на момент считывания не связано с каким-либо внешним признаком. Поэтому записываемые данные объединяют в блоки, обрамляемые специальными символами конца и начала (файлы). Прием данных из файлового ЗУ начинается после обнаружения приемником символа начала блока.

В циклических ЗУ слова доступны одно за другим с постоянным периодом, определяемым емкостью памяти. К такому типу среди полупроводниковых ЗУ относится видеопамять (VRAM).

Считывание в обратном порядке свойственно стековым ЗУ, для которых реализуется дисциплина «последний пришел - первый вышел». Такие ЗУ называют буферами LIFO (Last In - First Out).

Время доступа к конкретной единице хранимой информации в последовательных ЗУ представляет собою случайную величину. В наихудшем случае для такого доступа может потребоваться просмотр всего объема хранимых данных.

Ассоциативный доступ реализует поиск информации по некоторому признаку, а не по ее расположению в памяти (адресу или месту в очереди). В наиболее полной версии все хранимые в памяти слова одновременно проверяются на соответствие признаку, например, на совпадение определенных полей слов (тегов - от англ. tag) с признаком, задаваемым входным словом (теговым адресом). На выход выдаются слова, удовлетворяющие признаку. Дисциплина выдачи слов, если тегу удовлетворяют несколько слов, а также дисциплина записи новых данных могут быть разными. Основная область применения ассоциативной памяти в современных ЭВМ - кэширование данных.

Технико-экономические параметры ЗУ существенно зависят от их схемотехнологической реализации. По этому признаку также возможна классификация ЗУ, однако удобнее рассматривать этот вопрос применительно к отдельным типам памяти [4].

3.5. Основные структуры запоминающих устройств

Многочисленные варианты ЗУ имеют много общего с точки зрения структурных схем, что делает рациональным изучение некоторых обобщенных структур с последующим описанием особенностей и запоминающих элементов для конкретных ЗУ.

Общность структур адресных ЗУ особенно проявляется для статических ОЗУ и памяти типа ROM. Для статических ОЗУ и памяти типа ROM характерны структуры 2D, 3D, 2DM и блочные структуры на их основе.

В структуре 2D (рис. 3.3) запоминающие элементы ЗЭ организованы в прямоугольную матрицу размерностью $M = k \times m$, где M - информационная емкость памяти в битах; k - число хранимых слов; m - их разрядность.

Дешифратор адресного кода DC при наличии разрешающего сигнала CS (Chip Select) активизирует одну из выходных линий, разрешая одновременный доступ ко всем элементам выбранной строки, хранящей слово, адрес которого соответствует номеру строки. Элементы столбца соединены вертикальной линией - внутренней линией данных (разрядной линией, линией записи/считывания). Элементы столбца хранят одноименные биты всех слов. Направление обмена определяется усилителями чтения/записи под воздействием сигнала R/W (Read - чтение, Write - запись).

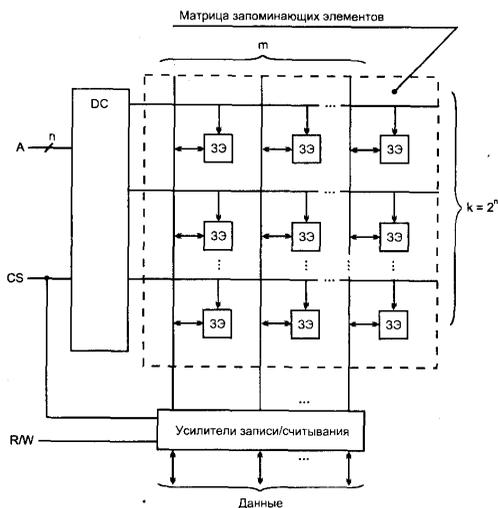


Рис. 3.3. Структура запоминающего устройства типа 2D

Структура типа 2D применяется лишь в ЗУ малой информационной емкости, т. к. при росте емкости проявляется несколько ее недостатков, наиболее очевидным из которых является чрезмерное усложнение дешифратора адреса (число выходов дешифратора равно числу хранимых слов).

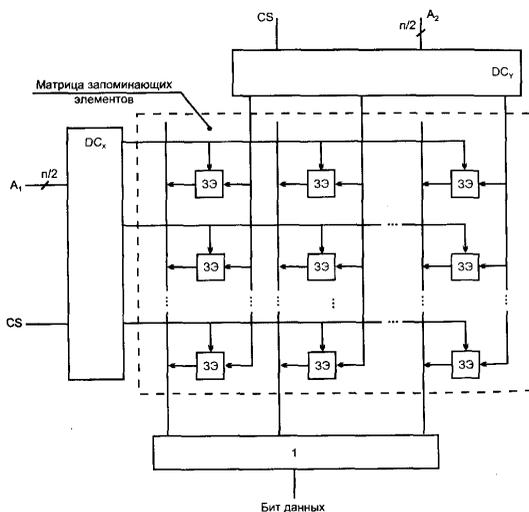


Рис. 3.4. Структура ЗУ типа 3D с одноразрядной организацией

Структура 3D позволяет резко упростить дешифраторы адреса с помощью двухкоординатной выборки запоминающих элементов. Принцип двухкоординатной выборки поясняется на примере ЗУ типа ROM (рис. 3.4), реализующего только операции чтения данных.

Здесь код адреса разрядностью n делится на две половины, каждая из которых декодируется отдельно. Выбирается запоминающий элемент, находящийся на пересечении активных линий выходов обоих дешифраторов [4].

4. МИКРОПРОЦЕССОРЫ И МИКРОКОНТРОЛЛЕРЫ

4.1. Основные определения. Структура и функционирование микропроцессорной системы

Необычайная популярность микропроцессоров объясняется тем, что их появление привело к внедрению вычислительной техники в самые разнообразные сферы жизни. Универсальность микропроцессоров ведет к большой тиражности их производства и, следовательно, к снижению их стоимости, а это расширяет круг потребителей и способствует дальнейшему удешевлению микропроцессоров.

Микропроцессором (МП) называют построенное на одной БИС/СБИС программно-управляемое устройство, осуществляющее процесс обработки информации и управления. Микропроцессоры появились, когда уровень интеграции ИС достиг значений, при которых блоки, необходимые для программной реализации алгоритмов, удалось разместить на одном кристалле. МП - центральный процессорный элемент микропроцессорной системы (микро - ЭВМ).

Основными частями микропроцессорной системы (МПС), являются микропроцессор, память, устройства ввода/вывода (внешние устройства) и интерфейсные схемы. Решаемая задача определяет реализуемую программу, структура микропроцессорной системы при решении различных задач остается неизменной, что и определяет ее универсальность.

Первый микропроцессор содержал 2300 транзисторов и имел рабочую частоту 108 кГц при среднем времени выполнения команды приблизительно десять тактов. Современный микропроцессор содержит более 200 млн транзисторов, работает на тактовых частотах более 5 ГГц, выполняет команды за один такт или даже за долю такта. Движущими силами столь впечатляющего развития служат все возрастающие потребности.

В процессе развития архитектура МПС претерпела существенные изменения. Первые МПС строились по так называемой Принстонской архитектуре (архитектуре фон Неймана), в которой память для команд и данных является общей. Эта архитек-

тура имеет свои достоинства - простоту, возможность оперативного перераспределения памяти между областями хранения команд и данных и др. Недостаток - последовательная во времени выборка из памяти команд и данных, передаваемых по одной и той же системной шине, что ограничивает производительность МПС. Тем не менее, в силу своих достоинств. Принстонская архитектура не только длительное время доминировала в микропроцессорной технике, но сохранила свое место и до настоящего времени.

В Гарвардской архитектуре память разделена на память команд и память данных, причем каждая из них имеет собственную шину для общения с процессором. При этом во время передач данных для выполнения текущей команды можно производить выборку и расшифровку следующей, что повышает производительность МПС. Реализация системы по сравнению с Принстонской архитектурой усложняется (в системе больше шин), ниже коэффициент использования памяти. Но в МПС высокой производительности и внутренних структурах высокопроизводительных МП Гарвардская архитектура находит широкое применение.

По другому архитектурному признаку, связанному с характером системы команд, микропроцессоры делятся на:

- CISC-процессоры;
- RISC-процессоры;
- VLIW-процессоры.

Процессоры CISC имеют так называемую сложную систему команд (CISC - Complex Instruction Set Computer), т.е. большой набор разноформатных команд при использовании многих способов адресации. Архитектура CISC присуща классическим процессорам, она в силу многообразия команд позволяет применять эффективные алгоритмы решения задач, но, в то же время, усложняет схему процессора и его стоимость и в общем случае не обеспечивает его максимального быстродействия.

RISC - процессоры имеют сокращенную систему команд (RISC - Reduced Instruction Set Computer), из которой исключены

редко применяемые команды. Форматы команд, по крайней мере, подавляющее их большинство, идентичны (например, все команды содержат по 4 байта), резко уменьшено число используемых способов адресации. Данные, как правило, обрабатываются только с регистровой или непосредственной адресацией. Увеличено число регистров процессора, т. е. его емкая внутренняя память, что позволяет редко обращаться к внешней памяти (модулю памяти МПС), а это повышает быстродействие системы. Идентичность временных циклов выполнения команд отвечает потребностям конвейерных схем обработки информации. В результате может быть достигнуто упрощение схемы процессора при увеличении его быстродействия.

VLIW - процессоры (VLIW - Very Long Instruction Word), особенностью которых является использование очень длинных команд (16 и более байт). Отдельные поля длинной команды определяют несколько подлежащих реализации операций, которые могут выполняться параллельно во времени в нескольких операционных устройствах процессора. Таким образом, одна длинная команда определяет сразу группу операций. VLIW - процессоры считаются перспективными для высокопроизводительных процессоров.

Эволюционное развитие микропроцессоров, направленное главным образом на повышение их производительности, обеспечивалось и обеспечивается несколькими факторами, в первую очередь следующими:

- применением конвейеров, т. е. разбиением на отдельные этапы выполняемых операций, которое позволяет переходить к выполнению этапов над следующими операндами сразу же после освобождения ступени конвейера от обработки предыдущих, т. е. не дожидаясь конца всей операции над предыдущими операндами;

- установлением кэш-памяти первого уровня на одном кристалле с процессором, что позволяет получить высокую скорость обмена процессора с кэш-памятью, а также применением ее как для команд, так и для данных;

- введением в структуру процессора предсказателей ветвления программы, повышающих эффективность кэш-памяти путем снижения процента промахов кэша;

- реализацией на одном кристалле с основным процессором вспомогательных специализированных процессоров (сопроцессоров), прежде всего для обработки данных с плавающей запятой, введением в структуру процессора дополнительных аппаратных средств реализации важных для решаемых задач специальных функций;

- выполнением команд не в порядке их следования в программе, а по мере готовности данных и ресурсов для данной команды с последующим учетом фактической последовательности выполнения команд;

- делением исполнительного ядра процессора на области (кластеры) с концентрацией по возможности операций внутри одного и того же кластера и минимизацией распространения сигналов на большие расстояния по кристаллу, поскольку времена распространения сигналов в пространстве уже становятся причиной ограничения быстродействия современных микропроцессоров;

- применением для внутренних исполнительных устройств процессора повышенной частоты тактирования с целью исключить возможные интервалы ожидания со стороны других устройств.

Перечисленные и некоторые новые намечающиеся способы повышения быстродействия микропроцессоров ведут к усложнению их структур.

Кроме понятий «микропроцессор» и «микропроцессорная система» существует также понятие «микропроцессорный комплект БИС», которое сейчас употребляется не слишком часто, но ранее имело широкое распространение. Микропроцессорным комплектом (МПК) называют совокупность БИС/СБИС, пригодных для совместного применения в составе МПС. Понятие МПК задает номенклатуру микросхем с точки зрения возможностей их совместного применения (совместимость по архитектуре, электрическим параметрам, конструктивным признакам и

др.). В состав МПК могут входить микросхемы различных серий и типов при условии их совместимости [4].

4.2. Структура микропроцессорной системы

Структура МПС является магистрально-модульной. В такой структуре имеется группа магистралей (шин), к которым подключаются различные модули (блоки), обменивающиеся между собой информацией по одним и тем же шинам поочередно, в режиме разделения времени.

Термин «шина» относится к совокупности цепей (линий), число которых определяет разрядность шины. Термин «шина» применяется также в более широком смысле как синоним термина интерфейс. Типична трехшинная структура МПС с шинами адресов (ША), данных (ШД) и управления (ШУ). Наряду с русскими терминами применяются английские АВ (Address Bus), DB (Data Bus) и СВ (Control Bus) или просто шины А, D, С. Трехшинная структура в чистом виде характерна для простых МПС, в более производительных системах шинная структура образует более сложную иерархию, но в своей основе разделение на перечисленные виды шин остается справедливым, хотя и может относиться не к отдельным шинам, а к их группам.

На рис. 4.1 показана типичная структура микропроцессорной системы. По адресной шине А в систему передаются адреса модулей, к которым обращается микропроцессор МП. В эту шину включен шинный формирователь ШФ, обеспечивающий работу МП на нагрузку, образуемую внешними цепями. Собственной нагрузочной способности у выводов МП, как правило, не хватает.

Адреса используются блоками постоянной и оперативной памяти ПЗУ и ОЗУ (ROM и RAM), а также адаптерами и контроллерами, посредством которых микропроцессор общается с различными внешними устройствами (ВУ). При адресации памяти старшие и младшие разряды адресов используются по-разному. Старшие разряды поступают на селектор адреса САзу. Соответственно старшим разрядам селектор разрешает работу

тех или иных блоков с помощью сигналов выбора кристаллов CS (Chip Select). Младшие разряды адресов используются для адресации слов внутри выбранного модуля. При работе оперативной памяти она получает от процессора управляющий сигнал записи/чтения R/W (Read/Write), указывающий на требуемое направление передачи данных между памятью и процессором. Постоянная память в подобном сигнале не нуждается, т. к. Используется в рабочем режиме только для чтения. Селектор адресов внешних устройств $CA_{ВУ}$ декодирует адреса, присвоенные адаптерам и контроллерам, и вырабатывает сигналы CS разрешения работы выбранным модулям. С внешними устройствами, работающими с параллельными кодами, процессор общается с помощью программируемого параллельного адаптера ППА. Показанный на рисунке адаптер ППА имеет три канала связи с внешними устройствами (т. е. может обслуживать до трех полноразрядных ВУ). Для каналов ППА могут быть выбраны различные режимы ввода, вывода и двунаправленных передач при разных способах обмена с процессором. Общение процессора с ВУ, работающими с последовательными кодами, производится с помощью программируемого связного адаптера ПСА, который преобразует параллельные коды, получаемые от процессора, в последовательные для ВУ или для последовательной линии связи, либо наоборот, получая последовательные коды, преобразует их в параллельные для передачи в процессор. Программирование адаптера позволяет настраивать его на разные протоколы и режимы обмена. Последовательные выходные сигналы адаптера ПСА поступают на линию передатчика TxD (Tranceiver Data), последовательные входные - на линию приемника RxD (Receiver Data).

Контроллер прямого доступа к памяти КПД (DMA, Direct Memory Access) обслуживает процесс прямого обмена данными между внешними устройствами и памятью. При отсутствии такого режима, называемого прямым доступом к памяти, передача данных между указанными модулями (например, между винчестером и ОЗУ) для каждого слова протекает так: сначала слово читается процессором из источника данных, потом оно записывается процессором в их приемник, что требует двух машинных циклов и замедляет передачу. При прямом доступе к памяти процессор отключается от шин системы и передает управление шинами предварительно запрограммированному КПД, который реализует более быструю непосредственную (за один цикл) передачу данных между источником и приемником. Особенно эффективен ПДП при блочных передачах.

Программируемые интервальные таймеры ПИТ (PIT, Programmable Interval Timer) выполняют над временными интервалами операции, играющие большую роль в ряде разнообразных ситуаций (часы реального времени, генерация звуковых сигналов, сторожевые таймеры, генерация временных меток в многозадачных процессах с разделением времени между задачами, выработка широтно-модулированных импульсных сигналов и т. д.). На рисунке показан таймер с тремя выходными каналами, каждый из которых может выполнять свою функцию по формированию временных последовательностей.

Кроме сигналов, индивидуально обозначенных на рис. 4.1, адаптеры и контроллеры имеют и многие другие управляющие сигналы, объединенные на рисунке в группы «У, А» (управление, адресация). В число таких сигналов входят стробы записи (для загрузки в программируемые модули управляющих слов), чтения (для чтения программой слов состояния модулей), адресные коды для адресации внутренних регистров модулей, сигналы сброса, тактирования и др.

Микропроцессор МП также вырабатывает множество сигналов управления системой, не показанных на упрощенной схеме (см. рис. 4.1). Передачи данных в МПС осуществляются по системной ШД, разрядность которой определяет понятие

«разрядность процессора». Эти передачи двунаправлены в зависимости от сигнала T (Transmit). При активном состоянии формируемого процессором сигнала чтения RD (Read) данные передаются через ШФ справа налево, при пассивном - в обратном направлении. К шине данных подключены информационные выводы всех модулей МПС. Выводы XI и X2 служат для подключения кварцевого резонатора или иных контуров, задающих частоту тактовому генератору, расположенному в МП. Вход RESIN является входом асинхронного сброса, приводящим МП в исходное состояние. Сигнал сброса L-активный. Сброс может быть осуществлен замыканием ключа К и автоматически происходит при включении питания U_{cc} . В этом случае благодаря цепочке RC напряжение на входе RESIN после включения питания нарастает постепенно, и в течение некоторого времени остается низким (ниже порогового), что равноценно подаче на этом интервале времени L-активного сигнала RESIN. Выполняя программу, МП обрабатывает команду за командой. Команда задает выполняемую операцию и содержит сведения об участвующих в ней операндах. После приема команды происходит ее расшифровка и выполнение, в ходе которого МП получает необходимые данные из памяти или внешних устройств. Ячейки памяти и внешние устройства (порты) имеют номера, называемые адресами, которыми они обозначаются в программе. Таким образом, по однонаправленной адресной шине МП посылает адреса, определяя объект, с которым будет обмен, по шине данных (двунаправленной) обменивается данными с модулями (блоками) системы, по шине управления в разных направлениях передаются управляющие сигналы. ПЗУ (ROM) хранит фиксированные программы и данные, оно является энергонезависимым и при выключении питания информацию не теряет. ОЗУ (RAM) хранит оперативные данные (изменяемые программы, промежуточные результаты вычислений и др.), является энергозависимым и теряет информацию при выключении питания (если не применяются специальные методы, например, автоматическое подключение автономного резервного питания на время отсутствия основного). Для приведения системы в работоспо-

собное состояние после включения питания ОЗУ следует загрузить необходимой информацией.

Устройства ввода/вывода (УВВ) или внешние устройства (ВУ) - технические средства для передачи данных извне в МП или память, либо из МП или памяти во внешнюю среду. Для подключения ВУ необходимо привести их сигналы, форматы слов, скорость передачи и т. п. к стандартному виду, воспринимаемому данным МП. Это и выполняется адаптерами и другими интерфейсными блоками [4].

4.3. Микроконтроллеры. Основные сведения

Микроконтроллеры (МК) - разновидность микропроцессорных систем, ориентированных на реализацию алгоритмов управления техническими устройствами и технологическими процессами. Микроконтроллеры проще, чем универсальные микро-ЭВМ. Микроконтроллеры - БИС такой функциональной законченности, которая позволяет решать в полном объеме задачи определенного класса с помощью одного кристалла.

Отличием МК от МПС является малый объем памяти и менее разнообразный состав внешних устройств. В состав универсальной микро-ЭВМ входят модули памяти большого объема и высокого быстродействия, имеется сложная иерархия ЗУ, поскольку многие задачи (автоматизированное проектирование, компьютерная графика, мультимедийные приложения и др.) без этого решить невозможно. Для МК ситуация иная, они реализуют несложные алгоритмы, и для размещения программ им требуются емкости памяти, на несколько порядков меньшие, чем у микро-ЭВМ широкого назначения. Для хранения промежуточных Данных достаточна память небольшой емкости. Набор внешних устройств также существенно конкретизируется и сужается, а сами они значительно проще. В результате модули универсальной микро-ЭВМ (процессор, память, интерфейсные схемы) требовалось выполнять как конструктивно самостоятельные, тогда как МК размещается на одном кристалле, хотя и имеет модули того же функционального назначения.

Сопоставляя микропроцессор (т. е. центральный процессорный элемент сложной системы) и МК с точки зрения коммерческих потребностей, можно видеть преобладание МК. Число пользователей МК в несколько раз превышает число пользователей микросхем МП. Применение МК поддерживается такими областями массового производства, как бытовая аппаратура, станкостроение, автомобильная промышленность, военное оборудование и т. д. Годовой мировой выпуск микроконтроллеров сейчас оценивается цифрой 2 млрд, а их номенклатура насчитывает тысячи типов.

Первые МК были выпущены фирмой Intel в 1976 г. (восьмиразрядный МК 8048). В настоящее время многими поставщиками выпускаются восьми-, 16- и 32-разрядные МК с емкостью памяти программ до десятков килобайт, небольшими ОЗУ данных и набором таких интерфейсных и периферийных схем, как параллельные и последовательные порты ввода/вывода, таймеры, аналого-цифровые и цифроаналоговые преобразователи, широтноимпульсные модуляторы и др.

Несмотря на появление новых 16- и 32-разрядных МК, наибольший успех на рынке остается за восьмиразрядными. Сейчас около половины рынка занято 8-разрядными МК, которые лидируют с большим отрывом относительно микроконтроллеров других разрядностей. На рынке восьмиразрядных микроконтроллеров доминирует следующая тройка: семейство 8051 фирмы Intel (аналоги микроконтроллеров этого семейства выпускаются несколькими фирмами), семейство AVR (фирмы Atmel) и микроконтроллеры семейства PIC (фирмы Microchip). В качестве примера современного микроконтроллера далее рассмотрена микросхема из семейства AVR. Сейчас самая большая доля рынка все еще принадлежит микроконтроллерам семейства 8051, за которыми следуют семейства AVR и PIC. Остальные микроконтроллеры по объему продаж значительно отстают от лидеров. Микроконтроллеру AVR предпочтение отдано как обладающему хорошо продуманной архитектурой и высоким быстродействием. Микросхемы AVR используют RISC - процессо-

ры, которые в последнее время интенсивно внедряются в структуры микроконтроллеров.

Микроконтроллеры марки AVR подразделяются на три семейства, среди которых базовым является семейство Classic. Ниже рассматривается представитель именно этого семейства, который для краткости называется просто микроконтроллером AVR.

Микроконтроллеры (далее иногда просто контроллеры) AVR имеют RISC - архитектуру и изготавливаются по усовершенствованной КМОП-технологии.

Контроллеры семейства AVR имеют следующие параметры:

- команды выполняются за один машинный такт, что при тактовой частоте 1 МГц дает производительность в 1 MIPS (Million Instructions Per Second);

- флэш-память программ емкостью 1 - 8 Кбайт имеет допустимое число репрограммирований 10^3 ;

- статическая память данных (SRAM) имеет емкость до 512 байт;

- память данных типа EEPROM с допустимым числом репрограммирований 10^5 имеет емкость 64 - 512 байт;

- многоуровневая система прерываний обслуживает от 3 до 16 источников запросов прерываний;

- имеется достаточно обширный набор периферийных устройств [4].

4.4. Структура микроконтроллера

МК AVR - восьмиразрядный RISC - микроконтроллер с Гарвардской архитектурой и пониженным энергопотреблением. Набор команд, ограниченность которого свойственна RISC - архитектурам, в данном случае необычно широк, однако при этом сохранено основное преимущество RISC - архитектур - повышенное быстродействие и сокращенное число операций обмена с памятью программ. Почти все команды размещаются в одной ячейке программной памяти и выполняются за один такт син-

хросигнала. Типичен режим с частотой синхронизации 1 МГц. Максимальная частота синхросигнала составляет 8 МГц. Доступ к памяти программ и памяти данных осуществляется через собственные шины этих модулей, поэтому можно не только сделать различными разрядности шин, но и реализовать параллелизм операций в процессах выполнения текущей команды и выборки и дешифрации следующей, т. е. ввести в работу МК элементы конвейеризации.

На рис. 4.2 показана схема МК AVR типа AT90S8515. Микроконтроллер имеет восьмиразрядную шину данных, посредством которой его модули обмениваются информацией.

Заметим, что с целью упрощения рисунка разрядности шин на нем не указаны, но они легко могут быть определены на основе достаточно простых соображений: все шины, выходящие на шину данных, имеют по 8 разрядов, такова же и разрядность блока регистров и АЛУ, счетчик команд соответственно емкости памяти программ (2048 слов, т.е. 4096 байт) является 12-разрядным, флэш-память имеет 16-разрядную организацию, определяющую разрядность регистра команд IR. Линии шины управления RESET, ALE, ICP, входы и выходы последовательных каналов блоков SPI и UART, линия канала последовательного программирования флэш-памяти и EEPROM, линии передачи аналоговых сигналов и др. являются одноразрядными.

Многие блоки AVR по назначению аналогичны рассмотренным в главе 5 блокам микропроцессора и имеют те же самые обозначения. Программный счетчик PC содержит адрес подлежащей выполнению команды и адресует флэш-память программ. Считанная из флэш-памяти команда поступает в регистр команд IR, ее КОП (код операции) декодируется дешифратором команд для выработки сигналов управления блоками микроконтроллера соответственно заданной операции, а КАД (адресная часть) адресует данные в блоке регистров или в памяти данных SRAM. В памяти типа EEPROM хранятся редко изменяемые данные (калибровочные константы и т. п.). Указатель стека SP используется для организации стека в некоторой области памяти SRAM, глубина стека ограничивается только наличием свобод-

ной области в этой памяти. Регистры общего назначения (РОН) объединены в регистровый файл.

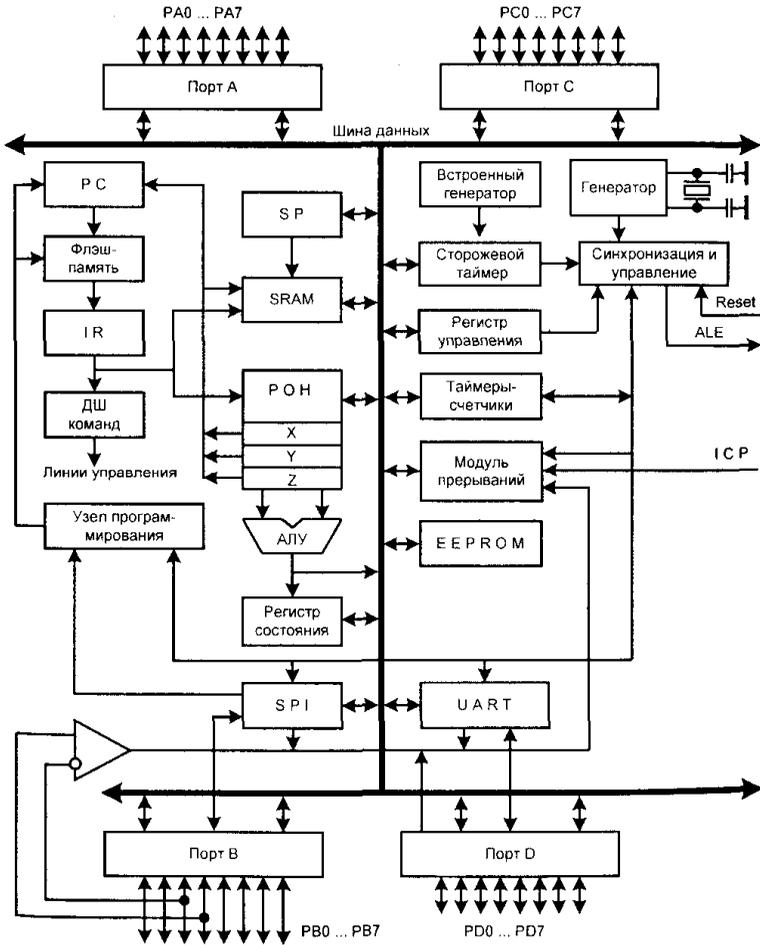


Рис. 4.2. Схема МК AVR типа AT90S8515

Арифметико-логическое устройство (АЛУ) способно выполнять операции над содержимым любой пары регистров блока и направлять результат в любой регистр, т. е. все регистры РОН непосредственно доступны для АЛУ. Этим микроконтроллер AVR отличается как от рассмотренного выше микропроцессора,

так и от микроконтроллеров других фирм, в которых рабочим регистром для АЛУ обычно служит только один регистр (аккумулятор). Наличие у АЛУ многих рабочих регистров позволяет выполнять операции за один такт. Три регистровые пары (X, Y, Z), получаемые объединением двух восьмиразрядных регистров в один 16-разрядный, используются для косвенной адресации. Регистр состояния по функциональному назначению аналогичен регистру флажков RF в структуре рассмотренного ранее микропроцессора, он содержит признаки результатов при выполнении некоторых команд (нуль, знак, перенос, половинный перенос и т. д.).

Генератор синхросигналов имеет внешние выводы для подключения кварцевого или иного резонатора либо внешнего тактирующего сигнала. Кроме основного синхрогенератора микроконтроллер имеет и дополнительный встроенный RC-генератор с фиксированной частотой 1 МГц (при напряжении питания 5 В) для тактирования сторожевого таймера. Вход RESET (L-активный) служит для сброса микроконтроллера (приведения его в исходное состояние), а также перевода его в режим программирования при подаче на этот вход специального повышенного напряжения 12 В. Выход ALE имеет то же назначение, что и одноименный выход рассмотренного ранее микропроцессора и используется при подключении к микроконтроллеру внешнего ЗУ. В этом случае реализуется режим мультиплексируемой шины: старший полуадрес выводится в течение всего цикла через один восьмиразрядный порт, а младший - через второй восьмиразрядный порт загружается в начале цикла во внешний регистр-зашелку, где сохраняется на все время цикла. После загрузки внешнего регистра-зашелки этот же порт используется для передачи данных и сигналов управления [4].

Модуль прерываний служит для приема и обработки запросов прерывания основной программы, как внутренних, так и внешних. Предусмотрено наличие 10 внутренних и двух внешних запросов.

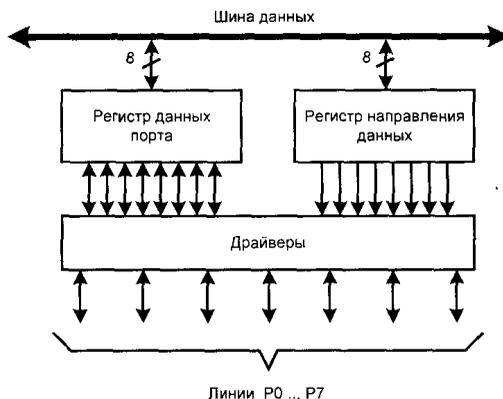


Рис. 4.3. Структура порта ввода/вывода микроконтроллера AVR

Структура одного из четырех портов ввода/вывода МК AVR приведена на рис. 4.3. Порт представляет собою набор из восьми линий. Каждая из восьми линий любого порта конфигурируется как входная или выходная индивидуально с помощью управляющего слова, загружаемого в регистр направления передачи. Каждый бит этого слова задает конфигурацию своей линии. Выводимые или вводимые данные поступают в регистр данных. Входные и выходные сигналы проходят через буферные каскады (драйверы) [4].

5. МЕТОДЫ И СРЕДСТВА ПРОЕКТИРОВАНИЯ СИСТЕМ НА КРИСТАЛЛЕ

5.1. Система на кристалле. Общие сведения

Системы на кристалле (СнК) – это методология разработки заказных микросхем на основе уже готовых сложнофункциональных блоков (IP-модулей). Не все заказные схемы разрабатываются в соответствии с этой методологией. Изделия с относительно небольшим числом элементов (до одного миллиона) или с предельными требованиями к электрическим и эксплуатационным параметрам целесообразно разрабатывать по традиционному маршруту одним коллективом разработчиков. Сложнофункциональные блоки, разработанные для традиционных заказных интегральных микросхем, могут быть использованы повторно при проектировании систем на кристалле. Производственной основой современных СнК является субмикронная КМОП – технология. Предельное быстродействие в СнК обычно требуется для синхрогенераторов и блоков ввода – вывода информации. Предварительную оценку возможностей техпроцесса можно сделать на основе результатов завершенных разработок.

Требуемые технические показатели системы на кристалле могут ограничиваться разными факторами: возможностями полупроводниковой технологии, конструкцией корпуса, условиями теплоотвода в аппаратуре и другими [2].

В большинстве случаев каждый из этих ограничивающих факторов можно изменить путем перехода на более дорогие технические решения. Например, уменьшить конструктивно - технологические размеры физической структуры кристалла, использовать многокристальные модули и принудительное охлаждение корпусов. Однако себестоимость продукции при этом может возрасти в несколько раз и сделать изделие неконкурентоспособным. Поэтому предельные параметры СнК рассматриваются в рамках конкретных конструктивно – технологических и экономических ограничений. В такой ситуации существует опасность, что требования, установленные в техническом задании, не могут быть выполнены используемыми средствами.

В традиционном маршруте проектирования все блоки разрабатываются заново и оптимизируются для конкретного применения. В маршруте СнК блоки отбираются по принципу совместимости без оптимизации их параметров для данного проекта.

Все IP-модули можно разделить на три основных класса: программные, параметризуемые и фиксированные. Программные блоки описаны на языке высокого уровня и всегда требуют доработки для каждого конкретного проекта. Параметризуемые блоки включают избыточные части, используемые для окончательной настройки. Фиксированные IP-модули оптимизированы для конкретного применения и наилучшим образом подходят для тех частей СнК, в которых требуются предельные параметры.

Несмотря на отличия в подходах к разработке составных блоков маршруты проектирования СнК и традиционных заказных БИС включают одни и те же основные этапы.

Проектирование систем на кристалле является универсальной и многоплановой дисциплиной, объединяющей в себе методы проектирования законченных аппаратно-программных комплексов, встраиваемых систем на основе стандартных процессоров и процессорных ядер, разработки встроенного программного обеспечения, программируемых (ПЛИС), полузаказных и заказных интегральных схем.

В общем случае, система на кристалле может включать в себя различные типы блоков: программируемые процессорные ядра, блоки программируемой логики, памяти, периферийных устройств, аналоговые компоненты и различные интерфейсные схемы. Не обязательно все такие блоки должны быть физически реализованы на одном кристалле: процессоры, блоки памяти, ПЛИС или СБИС могут использоваться как отдельные компоненты. В рамках излагаемой методологии ставится задача спецификации, верификации и оптимизации системы в целом, а реализация отдельных устройств в контексте проектирования рассматривается как проектирование ее составных частей.

Данный подход наряду с системами применим и к проектированию отдельных ПЛИС, в которых интегрируется все большая и большая функциональность, включая процессоры, память, блоки цифровой обработки сигналов, высокоскоростные входы/выходы и ряд других сложных IP-блоков.

Таким образом, тенденция к сближению архитектур ПЛИС, СБИС и систем с использованием стандартных компонентов в условиях быстро изменяющихся технологических возможностей и экономических факторов, предъявляет требования к унификации маршрутов проектирования.

5.2. Маршрут проектирования систем на кристалле

Общий маршрут проектирования систем на кристалле состоит из следующих основных этапов:

- концептуальное проектирование системы; основной задачей данного этапа является исследование проектируемой системы и получение ее исполняемых спецификаций на языке высокого уровня (стандартно на C/C++);

- проектирование, то есть трансформация исполняемой спецификации проекта на уровень регистровых передач (получение спецификаций на языках Verilog/VHDL) и далее на вентильный уровень;

- верификация проекта, то есть проверка проекта и проектных решений на соответствие исходной спецификации и другим требованиям в процессе проектирования и детализации;

- физическое проектирование, начиная от выбора технологического и библиотечного базиса и заканчивая получением финального описания проекта в формате GDSII.

При проектировании систем на кристалле концептуальный уровень является критическим для оценки общих характеристик системы. На этом уровне создается общая исполняемая спецификация проектируемой системы, позволяющая исследовать и оценить различные варианты ее построения и выбрать оптимальное решение, которое будет реализовано в дальнейшем. Здесь решаются следующие задачи:

- создание функциональной модели системы, то есть описание системы с точки зрения тех алгоритмов и функций, которые она должна выполнять, без привязки к способам их реализации;

- моделирование системы в ее операционной среде с реальными данными и сигналами (аудио- и видеоинформацией, радиоканалами, расположением и движением объектов и др.)

- определение архитектуры системы с точки зрения необходимых ресурсов и их организации для программно-аппаратной реализации функциональной модели.

Таким образом, имея исполняемую спецификацию системы, поведенческие модели и общую архитектуру, проектирование, верификация и топологическая реализация системы далее ведутся параллельно [2].

5.3. Концептуальный уровень проектирования

Основной целью в процессе спецификации проекта является определение и спецификация основных функций системы и создание исполняемой системной модели. С использованием моделирования, эта системная модель используется для верификации корректности работы системы с функциональной точки зрения в операционной среде, в которой она должна работать, а также для определения необходимых аппаратных ресурсов для работы и архитектуры системы. Общий маршрут проектирования на данном этапе приведен на рис. 5.1.

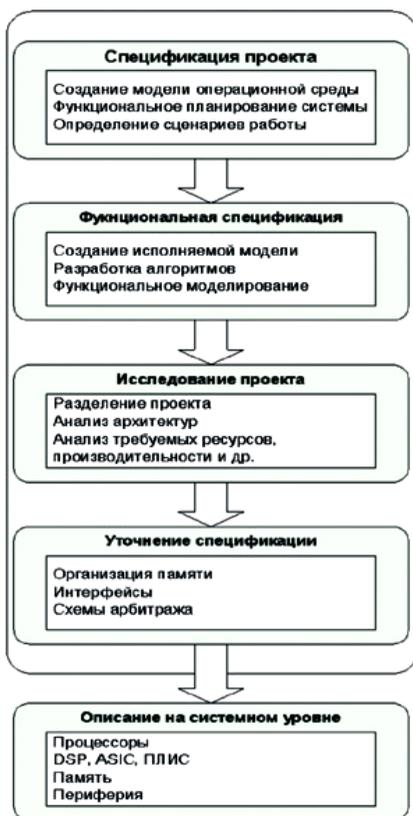


Рис. 5.1. Маршрут проектирования на концептуальном уровне

На этапе общей спецификации проекта определяется операционная среда, в которой должна работать система, основные сценарии работы, общие функциональные характеристики и протоколы.

Далее создается функциональная спецификация системы, целью которой является определение и моделирование функционирования системы с точки зрения выполняемых алгоритмов. Здесь может быть задано и промоделировано поведение всей системы в целом или ее отдельных блоков. Как правило, на этом уровне функции системы моделируются с реальными данными и сигналами.

На этапе исследования проекта моделируемые функции трансформируются и разделяются для выполнения на ряде платформ, или архитектур, которые содержат различные наборы компонентов, такие как программируемые процессоры, память, СБИС, ПЛИС или блоки системы на кристалле. Используя различные виды оценок, целью здесь является нахождение оптимальной архитектуры, которая отвечает заданным критериям, таким как работа в реальном времени, производительность, стоимость, потребляемая мощность. Программные функции оцениваются с точки зрения размера кода и наихудшего времени выполнения, измеряемого количеством тактов процессора, а аппаратные функции - в количестве эквивалентных вентиляей.

Производится уточнение спецификации системы, где создается более детальное описание системной архитектуры, которая передается на проектирование. Такое описание на системном уровне может содержать некоторые детали последующей реализации, но функциональная часть состоит из поведенческих моделей на языках C/C++/SystemC. Далее уже используется совместное программно-аппаратное проектирование с применением моделей конкретных процессоров и шин (функциональных моделей), блоков, описанных на языках проектирования аппаратуры VHDL/Verilog.

5.4. Проектирование и функциональная верификация

Функциональная верификация занимает все более важное место в общем маршруте проектирования. Если раньше под проектированием понималась разработка проекта на уровне регистровых передач (и далее переход на вентильный уровень средствами логического синтеза), а верификация проводилась средствами логического моделирования, то сейчас верификация начинается на поведенческом уровне на стадии разработки общей спецификации проекта.

Основными требованиями, предъявляемыми к составу средств функционального проектирования и верификации, являются:

- анализ архитектуры, производительности и других системных параметров проектируемых систем;
- проектирование аппаратно-программных систем, то есть возможность совместной разработки и верификации аппаратуры и встроенного программного обеспечения;
- проектирование систем с использованием процессорных блоков, то есть использование моделей процессоров при разработке аппаратуры и программного обеспечения;
- единая среда проектирования, от системного уровня до уровня регистровых передач и вентиляльного уровня с поддержкой языков C, C++, SystemC уровней 1.0 и 2.0 и языков описания аппаратуры Verilog и VHDL;
- наличие библиотек и высокоуровневых конструкций для функциональных блоков и коммуникационных каналов, включая таблицы связности;
- средства управления данными и документирования проектов.

На рис. 5.2 представлена схема маршрута проектирования и верификации систем на кристалле. Основные этапы маршрута проектирования систем на кристалле:

1. RTL-кодирование – разработка функционального описания блока на языках VHDL или Verilog - может выполняться как в ручном, так и в автоматизированном режимах.

2. Для моделирования используется тот же набор программных средств, что и при RTL-кодировании.

3. Логический синтез – процесс автоматизированного создания электрической (логической) схемы на базе RTL-описания и библиотек элементов логического уровня от производителя.

4. Вентильная верификация обычно сводится к статическому временному анализу списка цепей, полученному в результате логического синтеза. В отдельных случаях, когда размерность списка цепей невелика, можно выполнять моделирование на вентиляльном уровне.

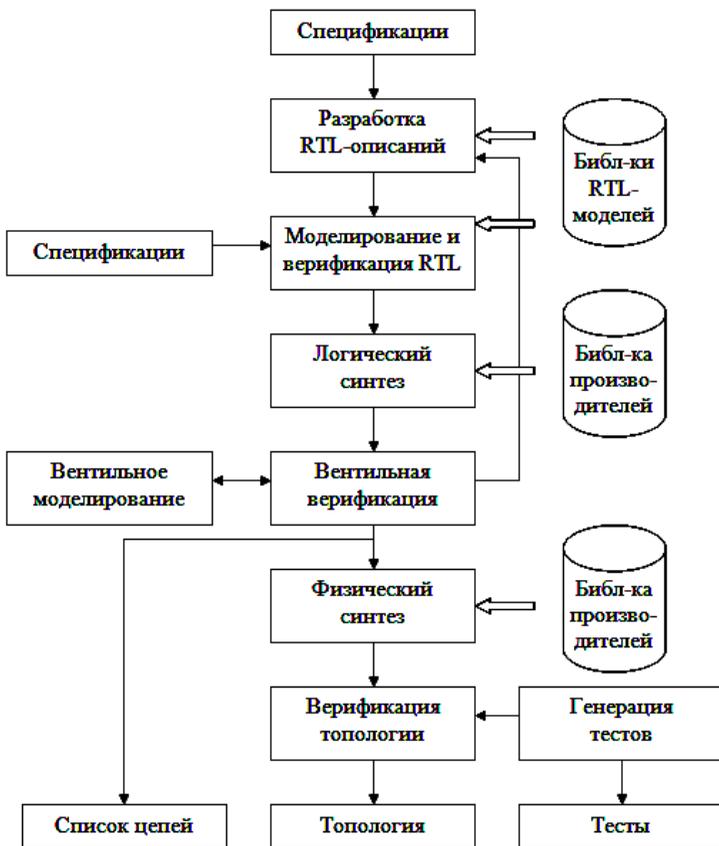


Рис. 5.2. Маршрут проектирования систем на кристалле

5. Целью физического проектирования является разработка топологии кристалла интегральной микросхемы при выполнении проектных норм и требований спецификации.

6. Основная цель функциональной верификации – комплексная отладка функциональной модели совместно с программным обеспечением. Обычно, функциональная верификация не может быть выполнена только средствами САПР. Для этого не хватает времени и вычислительных ресурсов. Совместно с программной верификацией выполняется и эмуляция сис-

темы с использованием специальных макетов. Функциональная верификация проводится совместно с функциональным проектированием и составляет с ним единый итерационный цикл [2].

На выходе маршрута должны быть представлены: список цепей (Verilog или VHDL), производственные тесты или топология в формате GDSII.

ЗАКЛЮЧЕНИЕ

Материал пособия собран из различных источников и представлен в виде краткого обзора некоторых вопросов проектирования специализированных интегральных схем и систем на кристалле. Рассмотрены структуры и принципы работы основных функциональных узлов комбинационного и последовательностного типов. Отдельные главы посвящены запоминающим устройствам и микроконтроллерам. Описаны структуры микропроцессорных и микроконтроллерных систем. Особое внимание уделено основным структурам статических оперативных запоминающих устройств (ОЗУ), так как их проектирование на базе программируемых интегральных схем вынесено на лабораторный практикум.

Учебное пособие необходимо студентам направления 12.03.01 «Приборостроение» при изучении лекций по дисциплине «Специализированные БИС и устройства функциональной электроники в приборостроении», а также при подготовке к лабораторным и практическим занятиям.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Музылева, И.В. Элементная база для построения цифровых систем управления [Текст] : учеб. пособие / И.В. Музылева, – М.: Техносфера, 2006. – 144 с.
2. Немудров, В. Д. Системы на кристалле. Проектирование и развитие [Текст] / В. Д. Немудров, Г.В. Мартин – М.: Техносфера, 2004. – 246 с.
3. Новиков, Ю.В. Введение в цифровую схемотехнику [Текст] : учеб. пособие / Ю.В. Новиков. – М: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. – 343 с.
4. Угрюмов, Е.П. Цифровая схемотехника [Текст] : учеб. пособие / Е.П. Угрюмов. – 2-е изд., перераб. и доп. – СПб. : БХВ-Петербург, 2004. – 800 с.

ОГЛАВЛЕНИЕ

Введение	3
1. Интегральные устройства микроэлектроники.....	4
1.1. Основные определения и понятия	4
1.2. Функции цифровых устройств	8
1.3. Простейшие и сложные логические элементы	9
1.4. Модели описания работы цифровых устройств	18
1.5. Основные обозначения устройств и функциональных узлов на схемах	23
1.6. Серии цифровых микросхем	26
2. Основные функциональные узлы интегральных схем.....	30
2.1. Функциональные узлы комбинационного и последователь- ностного типа	30
2.2. Дешифраторы и шифраторы	31
2.3. Мультиплексоры и демультимплексоры	33
2.4. Компараторы кодов	36
2.5. Триггерные устройства. Классификация. Основные сведения	38
3. Запоминающие устройства.....	43
3.1. Основные сведения. Система параметров. Классификация	43
3.2. Параметры запоминающих устройств	44
3.3. Входные и выходные сигналы запоминающих устройств ..	46
3.4. Классификация современных запоминающих устройств ...	47
3.5. Основные структуры запоминающих устройств	53
4. Микропроцессоры и микроконтроллеры.....	56
4.1. Основные определения. Структура и функционирование микропроцессорной системы	56
4.2. Структура микропроцессорной системы	60
4.3. Микроконтроллеры. Основные сведения	65
4.4. Структура микроконтроллера	67
5. Методы и средства проектирования систем на кристалле.	72
5.1. Система на кристалле. Общие сведения	72
5.2. Маршрут проектирования систем на кристалле	74
5.3. Концептуальный уровень проектирования	75

5.4. Проектирование и функциональная верификация	77
Заключение	81
Библиографический список	82

Учебное пособие

Пирогов Александр Александрович

ПРОЕКТИРОВАНИЕ ИНТЕГРАЛЬНЫХ СХЕМ
И ИХ ФУНКЦИОНАЛЬНЫХ УЗЛОВ

В авторской редакции

Подписано к изданию 08.12.2014.

Объем данных 810 Кб

ФГБОУ ВПО «Воронежский государственный
технический университет»

394026 Воронеж, Московский просп., 14