

В.Л.Артюхов,
Г.А.Копейкин,
А.Ашальто

Настраиваемые модули для управляющих логических устройств

Ленинград
Энергоиздат
Ленинградское отделение
1981

ББК 32.97
А 86
УДК 681.325.6

Рецензент В.Л.Генкин

Артюхов В.Л. и др.

А86 Настраиваемые модули для управляющих логических устройств/Артюхов В.Л., Копейкин Г.Н., Шальто А.А. – Л.: Энергоиздат. Ленингр.отд-ние, 1981. – 168 с., ил.

60 к.

Приведены новые методы построения и рационального использования настраиваемых логических модулей из функциональных элементов и элементов с двусторонней проводимостью. Найдены оценки сложности схемных реализаций. Предлагаемый подход распространен на цифровые интегральные схемы и релейно-контактные элементы, серийно выпускаемые промышленностью. Впервые рассматриваются вопросы унификации нерегулярных комбинационных схем.

Книга предназначена для специалистов в области проектирования логических элементов и устройств, а также для студентов и аспирантов соответствующих специальностей.

А $\frac{30502-115}{051(01)-81}$ 162-81(Э). 2405000000

ББК 32.97
6Ф7

Мы надеемся, что, прочитав эту книгу, читатель сможет:

- а) имея список требуемых логических формул, создать схему наиболее простого модуля, который путем настройки можно заставить «выполнять» любую формулу из этого списка;
- б) строить логические схемы из настраиваемых модулей, наиболее полно используя их функциональные возможности;
- в) объективно оценивать логическую эффективность логических элементов и их серий.

Те, кто сталкивается с подобными рода задачами в своей повседневной работе, вряд ли нуждаются в дополнительных разъяснениях и поэтому могут сразу приступить к изучению гл. 2. Однако мы рассчитываем и на интерес тех читателей, которые только начинают заниматься проблемами проектирования дискретных систем или испытывают затруднения в связи с появлением в дискретной технике массы новых сведений. Именно их вниманию предлагается материал, изложенный во введении в гл. 1.

Настоящая книга посвящена методам построения настраиваемых модулей, применяемых в управляющих логических устройствах. В более узком плане, речь в ней пойдет о модулях комбинационного типа и схемах преимущественно с параллельной обработкой информации.

У читателя - проектанта систем управления может возникнуть серия вопросов: «Мы в настоящее время создаем свои устройства на серийно выпускаемых элементах. Предлагаете ли вы перейти на какие-то неизвестные нам «настраиваемые модули»? Где эти модули приобрести? Какие преимущества они обеспечат?»

На самом деле «настраиваемые модули» - это не какие-то небывалые, новые элементы. Все проектанты стремятся к тому, чтобы не строить систему из элементов «россыпью», а применять заранее сделанные заготовки или сборки, которые и названы здесь модулями. В большинстве случаев модули реализуются из серийно выпускаемых промышленностью элементов и входят в состав микросборок, субблоков, каскадов. Модули имеют некоторое число дополнительных наружных выводов. Назначение этих выводов состоит в том, чтобы сделать схему более универсальной. А это, в свою очередь, необходимо для того, чтобы сократить номенклатуру схемно-конструктивных единиц и добиться их унификации.

В последнее время модули все чаще приобретают законченное конструктивное исполнение. Примером тому являются модули, разработанные по программе стандартизации радиоэлектронной аппаратуры ВМС США [50].

Авторы предвидят также другой вопрос: «Сейчас все шире внедряются вычислительные машины, а вы пишете о логических устройствах с параллельной обработкой информации. Может быть, лучше и проще перейти к использованию управляющих логических машин (УЛМ) или цифровых вычислительных машин (ЦВМ), чем строить дискретные специализированные схемы?»

По нашему мнению, программный и схемный принципы обработки информации еще долго будут сосуществовать, не отрицая, а дополняя друг друга. Область работы авторов, а именно промышленная автоматика, такова, что нам ближе асинхронные параллельные автоматы. На их базе получены основные результаты, излагаемые в книге. Однако это не исключает применения этих результатов при построении УЛМ, микропроцессоров и других устройств с последовательной обработкой информации ввиду того, что при их построении и использовании должны применяться те же основные принципы, что и для предлагаемых модулей: универсальность, максимальное использование функциональных возможностей, простота построения, многофункциональность и настраиваемость.

Цель приводимого ниже краткого обзора развития дискретной автоматки состоит в том, чтобы ввести читателя в круг проблем, связанных с созданием и применением настраиваемых логических модулей.

Дискретная, или релейная, автоматика существует очень давно. Не будем останавливаться на хитроумных механических приспособлениях, предложенных в древности. Отметим, что развитие дискретной автоматки, в современном смысле слова, началось с изобретения электромагнитного реле в первой половине XIX века. Релейно-контактные схемы нашли применение на автоматических телефонных станциях, на железных дорогах, во всевозможных устройствах защиты, пуска и блокировки. Их использование расширилось в 50-е годы нашего века, чему способствовали возросший спрос на автоматизацию различных промышленных процессов, успехи технологии, приведшие к созданию надежных малогабаритных реле, а также развитие теории синтеза контактных логических

сетей. Особую роль в СССР сыграли исследования члена-корреспондента АН СССР М. А. Гаврилова, чья книга [13] до сих пор является одной из наиболее полных и лучших в этой области.

В начале 60-х годов стала интенсивно развиваться бесконтактная релейная техника. К этому времени выяснилось, что ряд полупроводниковых и магнитных элементов ведут себя подобно контактам реле, т. е. могут находиться в двух стабильных состояниях: открытом и закрытом. При этом они значительно превосходили контактные элементы по быстродействию и допустимому числу переключений, а также по некоторым электрическим характеристикам.

В это время были созданы также известные промышленные серии бесконтактных логических элементов, такие, как «Логика-Т», магнитно-вентильные элементы и др. [45]. Дальнейшим шагом явилось создание микросхем и больших интегральных схем.

Однако ожидавшегося полного вытеснения контактных элементов не произошло вследствие преимуществ, которые они обеспечивали: простоты реализации схем, удобства стыковки с источниками информации и командоаппаратами, возможности коммутации мощных сигналов, простоты схем электропитания, наличия гальванической развязки цепей.

Сейчас все чаще говорят о смешанном элементном базисе, включающем в себя микросхемы для реализации логических операций, контактные элементы для ввода и вывода информации, выполнения ряда простых блокировок и гальванических развязок, а также ферритовые и полупроводниковые «объемные» элементы, осуществляющие длительное запоминание, промежуточное усиление и некоторые другие необходимые функции.

Микроминиатюризация элементов привела к дальнейшему росту объема автоматизации, так как появилась возможность создавать малогабаритные устройства программного управления, дискретные регуляторы, цифровые измерительные и преобразовательные схемы.

Если раньше ЦВМ строились на электровакуумных приборах, а системы дискретной автоматики - на реле, то с появлением бесконтактных логических элементов, и в особенности микросхем, создалась единая элементная база для построения ЦВМ и устройств промышленной автоматики. При этом одни и те же программы управления можно реализовать как с помощью специализированных асинхронных автоматов, так и посредством ЦВМ, причем их основные отличия состоят в методах программирования и выполнения логических операций. Если в асинхронных автоматах алгоритмы работы задаются схемами соединения логических элементов, а обработка информации ведется, как правило, параллельно, то в ЦВМ обычно используется некоторая универсальная схема соединения элементов. Алгоритм работы записывается в специальное программное устройство, а обработка информации ведется последовательно. Этот же принцип заложен в основу построения УЛМ и появившихся в последнее время микропроцессоров.

Со схемной точки зрения ЦВМ обладает более упорядоченной структурой, так как она содержит большое число устройств регулярного типа (регистры, счетчики, дешифраторы и т. д.) [6].

Проигрывая в гибкости и универсальности, специализированные автоматы обладают рядом ценных преимуществ по сравнению с ЦВМ: большим быстродействием при решении крупных массивов логических уравнений и меньшими габаритами, если выполняемые функции не слишком сложны.

Следует отметить также, что применение параллельных структур в промышленной автоматике позволяет в большинстве случаев обеспечить значительный выигрыш по надежности и долговечности. Действительно, объекты автоматизации часто содержат большое число исполнительных механизмов (*ИМ*), работающих взаимосвязано, но не одновременно. Если в ЦВМ или УЛМ один и тот же блок (например, процессор) обслуживает все *ИМ*, то в рассматриваемых автоматах эта нагрузка распределена между отдельными параллельными цепями. При этом у каждой из них ответственность меньше, а временной режим легче.

Представляется вероятным, что последовательный и параллельный принципы обработки информации будут и в дальнейшем применяться в сочетании друг с другом.

Область промышленной автоматики развивалась в соответствии с веяниями времени. В ней были и повальное увлечение бесконтактной техникой, и возврат к контактной, и споры вокруг микроэлектроники, и борьба между телемеханическими и радиальными принципами передачи информации, и, конечно, между параллельной («зачем делать сложно то, что можно сделать просто?») и последовательной («зачем делать по-разному то, что можно сделать одинаково?») структурами логических устройств.

Однако даже в самых противоположных технических решениях сохранялись с замечательным постоянством две тенденции развития: а) от схем из малых компонентов - к схемам повышенного уровня функциональной и конструктивной интеграции; б) от индивидуальных устройств - к

устройствам широкого применения. Указанные тенденции наблюдались на каждом этапе совершенствования элементной базы, структур и принципов построения систем дискретной автоматики.

Обратимся к примерам. В начале 60-х годов создавался набор магнитно-вентильных элементов [45]. Каждая элементарная ячейка представляла собой инвертор, выполняющий функцию «2 ИЛИ - НЕ». Известно, что, имея в номенклатуре единственный тип ячейки «2 ИЛИ - НЕ», можно, пользуясь теоремой о функциональной полноте [43], построить любую комбинационную схему. Поэтому казалось заманчивым обойтись ячейкой одного типа, и подобное предложение было подробно проработано.

Рассматривался также другой вариант - заранее сформировать из элементарных ячеек более крупные элементы, например «2 ИЛИ», «3 ИЛИ», «2 И», «3 И», «Запрет» и др. Это расширяло номенклатуру и приводило к сокращению межэлементного монтажа и уменьшению габаритов проектируемых устройств.

Для сокращения номенклатуры можно пойти на некоторую избыточность, вспомнив, что один крупный элемент может заменить несколько типов более мелких элементов. В частности, в магнитно-вентильной серии был предложен элемент, реализующий функцию

$$y = (x_1 \vee x_2 \vee x_3)(x_4 \vee x_5)x_6 x_7 x_8.$$

Выбор функции, описывающей элемент, оказался удачным, так как такой элемент реализует большое число подфункций от числа переменных, меньших восьми. Чтобы получить эти подфункции, необходимо на некоторые входы подать константы 0 или 1 либо приравнять входные сигналы друг другу. Элемент с приведенной выше функцией являлся «многофункциональным» или «настраиваемым», хотя в то время эти термины почти не употреблялись.

При использовании микроэлектронной технологии также существуют три варианта выбора номенклатуры серии: а) один элемент «2 И - НЕ»; б) несколько специализированных элементов повышенной интеграции; в) один или несколько многофункциональных или настраиваемых элементов. В этом случае преимущества от повышения уровня интеграции еще выше, так как технология межвентильного монтажа в пределах корпуса значительно более совершенна по сравнению с технологией межкорпусного монтажа.

Однако с повышением уровня интеграции число типов элементов стремительно растет. Воспользуемся такой аналогией: предположим, что завод, выпускающий типографские шрифты, состоящие из отдельных букв, захочет делать целые слова. Тогда ему вместо 33 наименований продукции придется выпускать десятки и сотни тысяч наименований (в таком положении находится, кстати, производство шрифтов в странах, пользующихся иероглифами). Повысим уровень интеграции еще больше - до фраз, состоящих из нескольких слов. С такой номенклатурой уже невозможно справиться никакому производству.

С логическими элементами получается подобная картина. Известно, что число функций от n переменных составляет

$$\phi(n) = 2^{2^n}$$

Это означает, что, например, при пяти входах может существовать

$$\phi(5) = 2^{32}$$

различных элементов. Вот и попробуйте угадать, какие из них стоит запускать в серийное производство. Проблема схемной интеграции логических устройств является очень важной, так как ее неудовлетворительное решение препятствует полному использованию достижений технологии. Мы можем в одном корпусе разместить сложнейшую схему, но не знаем какую, и в итоге в большинстве случаев размещаем в нем несколько простых не связанных между собой вентиляей!

Поиски решения указанной проблемы ведутся в разных направлениях, и в том числе на путях, предусматривающих пересмотр традиционных методов построения систем управления. Но все предложения в своей основе используют многофункциональные или настраиваемые модули либо в роли ячеек однородной среды, либо в качестве микропроцессоров, либо, наконец, в виде самостоятельных схем для построения более сложных устройств.

Иногда многофункциональные модули выпускаются промышленностью в виде микросхем, как это имеет место, например, в серии 108 [44, 51]. Чаще же они формируются из серийно выпускаемых элементов непосредственно предприятиями - разработчиками систем управления и конструктивно оформляются в виде микросборок или на платах печатного монтажа.

По сравнению со специализированными устройствами настраиваемые модули более универсальны, но обладают определенной избыточностью - при любой конкретной настройке часть их внешних выводов или внутренних элементов могла бы быть исключена.

Из сказанного ясно, в каких случаях принимается решение о применении (формировании) настраиваемых модулей. Делается это на основе рассмотрения структур создаваемых систем путем выделения в них схем массового и редкого применения. Первые из них целесообразно оставить специализированными, а вторые заменить небольшой номенклатурой модулей, настраиваемых на реализацию различных схем. Задача здесь состоит в том, как привести различные схемы «к общему знаменателю», т. е. найти такую «покрывающую» схему, из которой они получались бы путем настройки. Эта схема должна иметь минимальное число дополнительных выводов и простую реализацию, так как никому не нужно универсальное устройство, являющееся механической суммой объединяемых им схем.

Предположим теперь, что настраиваемый модуль создан. Возникает вопрос, как им пользоваться при синтезе различных сложных устройств, так как при неправильном его применении очень легко растерять его богатые функциональные возможности. Опыт создания и применения настраиваемых модулей убедительно свидетельствует: вместе с модулями должна создаваться и четкая простая инструкция по их использованию. Занимаясь разработкой такого рода инструкции, авторы обнаружили, что она при годна не только для специально созданных настраиваемых модулей, но и для обычных, широко применяемых серий микросхем [51].

Попытаемся обобщить сказанное. Итак, схемы дискретной автоматики рациональнее строить из крупных модулей, чем из отдельных элементов. Для этого надо уметь разрабатывать такие модули и знать, как ими потом пользоваться. В одном модуле можно объединить целый ряд функций, если это не связано со слишком большой избыточностью. Такой настраиваемый модуль пригоден для построения самых различных логических схем. Наконец, умея пользоваться настраиваемыми модулями, можно обобщить эти методы и на элементы, не объединенные в модули. Насколько авторам удалось справиться с изложенными задачами, можно будет судить из последующих глав книги. Первая из них содержит более подробные сведения о структуре, составе, особенностях технической реализации и алгоритмах работы управляющих логических устройств. Вторая объясняет, какие именно настраиваемые модули надо формировать. Третья - как их формировать из функциональных элементов, с тем, чтобы модули получились хорошими по выбранным критериям. В четвертой же описано, как пользоваться этими модулями. В пятой все и (ложечное обобщено и распространено на цифровые интегральные микросхемы широкого применения. Шестая глава отдает дань одному из «экзотических» направлений в теории логиче-14 синтеза - однородным структурам. Седьмая посвящена модулям из элементов с двусторонней проводимостью - их построению и применению. К таким элементам относятся и электромагнитные реле, к которым, вопреки моде, авторы питают глубокую привязанность.

Материал книги формировался в результате работ авторов области проектирования управляющих логических устройств и настраиваемых модулей для их построения, а также на основе читаемых авторами курсов лекций на курсах повышения квалификации.

Авторы выражают искреннюю благодарность за внимание и помощь, оказанные им, докт. техн. наук, проф. И. В. Прангишвили, докт. техн. наук, проф. В. Г. Лазареву, докт. техн. наук, проф. В. А. Кондрашову, докт. техн. наук, проф. И. Р. Фрейдзону, канд. техн. наук Л. М. Фишману и канд. техн. наук Л. Я. Розенблюму. Авторы пользуются также случаем поблагодарить многочисленных слушателей курсов повышения квалификации за то, что они заставляли нас отвечать на такие вопросы, которые мы предпочли бы обойти, но без ответов на которые вряд ли родилась бы эта книга.

Замечания и пожелания по книге просьба присылать по адресу: 191041, Ленинград, Марсово поле, 1, Ленинградское отделение Энергоиздата.

Авторы

Логическое управление в системах автоматики

1-1. ЗАДАЧИ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ. СТРУКТУРА И ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ ЛОГИЧЕСКИХ УСТРОЙСТВ

Логическим называется управление с помощью сигналов, принимающих конечное число фиксированных значений, осуществляемое в соответствии с заданной программой (алгоритмом). Обычно используют двухуровневые, или двоичные, сигналы. Один уровень обозначают условно 0, а другой - 1.

Примерами управляющих логических устройств могут служить распространенные в промышленной автоматике системы пуска и останова различных агрегатов, автоматического аварийного управления и защиты, дистанционного управления, сигнализации и контроля [1, 54].

Наряду с использованием в составе больших систем, логические устройства применяются также и при решении разного рода частных задач, например, в специализированных преобразователях, обычно выполняющих алгебраические, геометрические или табличные преобразования.

Логические устройства могут существовать как в виде законченных приборов, так и входить в состав систем регулирования в сочетании с аналоговыми схемами, осуществляя те или иные переключательные функции.

Наиболее общая модель логического устройства, или автомата, имеет n двоичных входов (x_1, x_2, \dots, x_n) и m двоичных выходов (y_1, y_2, \dots, y_m), т. е. представляет собой (n, m) - полюсник. Сигналы, поступающие на входы автомата, образуют множество входных воздействий X , а сигналы, вырабатываемые на его выходах, - множество выходных реакций Y . Входные воздействия поступают от источников информации (*ИИ*), к которым относятся кнопки, ключи, тумблеры, а также сигнализаторы состояния внешней среды и управляемого объекта. Источниками информации могут являться также выходы других дискретных устройств.

Определение состава и характеристик *ИИ* должно основываться на глубоком изучении автоматизируемого процесса. При их выборе обычно, сознательно или интуитивно, руководствуются принципом максимального воздействия *ИИ* на автоматизируемый процесс. Иными словами, в первую очередь выбирают такие *ИИ*, изменение состояния которых наиболее явно и непосредственно сказывается на изменении состояний и **сполнительных механизмов** (*ИМ*). Особенно легко указанный принцип прослеживается на примере выбора органов управления на пульте оператора. Каждый ключ, кнопка, тумблер имеют надпись, характеризующую то действие, которое должно произойти при воздействии на них, если этому не помешают блокировки, осуществляемые системой автоматики. Аналогично и другие *ИИ* выбираются так, чтобы последствия их срабатывания можно было предвидеть.

Принцип максимального воздействия не всегда осуществим, так как ряд «ключевых», с точки зрения автоматизируемого процесса, параметров не может быть замерен непосредственно, например, вследствие отсутствия соответствующих приборов. В этих случаях приходится прибегать к косвенным измерениям или вычислениям, вводя вместо одного *ИИ* их совокупность, в целом дающую необходимую информацию. Иногда от принципа максимального воздействия сознательно отходят, если опасаются нарушений в управлении при ложном срабатывании одного или нескольких *ИИ*. В этих случаях прибегают к прямому либо косвенному дублированию входных данных. Наконец, в особенности в специализированных преобразователях, мы не всегда обладаем возможностью выбирать различные варианты задания входной информации. Однако в целом принцип максимального воздействия является довольно универсальным. Непосредственное следствие его применения состоит в простоте получающихся логических устройств, так как число элементов в схеме, *ИИ* которой выбраны по принципу максимального воздействия, зависит от числа входов по закону, близкому к линейному, в то время как сложность произвольной схемы определяется показательным законом [29].

Выходные реакции Y выдаются схемой на объекты управления - исполнительные механизмы, к которым относятся всевозможные пусковые устройства, средства сигнализации, запоминающие устройства и т. п. Основными характеристиками *ИМ* являются: позиционность и наличие либо отсутствие памяти.

Позиционность характеризует число устойчивых состояний, в которых может находиться *ИМ*. В дальнейшем мы будем рассматривать *ИМ*, имеющие два устойчивых состояния («открыто - закрыто», «включено - отключено»).

ИМ, не обладающий памятью, находится в активном (включенном) состоянии до тех пор, пока на его вход действует командный сигнал, и переходит в пассивное (отключенное) состояние при снятии этого сигнала. *ИМ*, обладающий памятью, сохраняет свое состояние после снятия управляющего воздействия. Для таких *ИМ* требуются два управляющих сигнала - «Пуск» и «Стоп» либо «Запись» и «Стирание», каждый из которых подается на свой вход. Разрешающий сигнал («Пуск», «Запись» и т. п.) будем обозначать индексом 1; например, сигнал y_{i1} означает команду на пуск i -го *ИМ*.

Запрещающий сигнал («Стоп», «Стирание» и т. п.) обозначим индексом 0; например, y_{i0} - команда на останов. При этом необходимо различать командные (разрешающие и запрещающие) и информационные сигналы (y'_{i1} и y'_{i0}), которые появляются на выходе сигнализаторов положения *ИМ* после обработки командных сигналов.

Автомат преобразует в соответствии с заданным алгоритмом набор входных воздействий X в набор выходных реакций Y . В зависимости от вида реализуемого алгоритма различают одноктактные (комбинационные) и многотактные (с памятью) автоматы. У первой разновидности автоматов выходная реакция в данный момент времени определяется только текущим значением набора входных воздействий и не зависит от того, какие сигналы и в каком порядке подавались на входы или были на выходах автомата в предшествующие моменты времени, а у второй - выходная реакция зависит также от предшествующих значений входных воздействий или выходных реакций.

Запоминание предшествующих значений аппаратно реализуется совокупностью обратных связей, содержащих элементы «память» ($Я$) и «выдержка времени» ($ВВ$). Эти обратные связи могут также замыкаться через объект управления или внешнюю среду.

Рассмотрим более подробно структуру систем логического управления. Важнейшей особенностью этой структуры является то, что схемы включения и отключения *ИМ* обычно совмещают с элементами памяти, которые выполняются при этом механическим путем. Это делается с той целью, чтобы при исчезновении напряжения питания не возникали аварийные ситуации. Таким образом, в системах рассматриваемого класса имеется память, распределенная по объектам управления, которая поэтому не может быть минимизирована [20]. В большинстве случаев для таких систем характерно также и то, что соответствующие им алгоритмы функционирования можно реализовать как при помощи автоматов с памятью, так и комбинационными схемами за счет введения входных сигналов от дополнительно устанавливаемых сигнализаторов положения и параметров [21]. В системах промышленной автоматики алгоритмы работы обычно реализуются вторым способом - путем почти повсеместного использования комбинационных схем. Для таких систем характерно, что комплекс «система управления - объект управления» может вести себя как многотактное устройство, несмотря на то, что собственно система управления практически одноктактна. Это объясняется наличием обратных связей, охватывающих систему управления совместно с объектом управления. Эти обратные связи содержат механические элементы (в частности, управляемую арматуру), которые исключают замыкание контуров обратной связи для электрических сигналов. Поэтому сигнализаторы положения рассматриваются в этих системах как собственно *ИИ* наряду с органами управления и сигнализаторами параметров.

Таким образом, можно сделать вывод, что для систем промышленной автоматики важнейшие задачи в области логического синтеза связаны с построением комбинационных схем. Однако построение этих схем в данном случае обладает рядом особенностей. Во-первых, основная проблема синтеза асинхронных автоматов, к классу которых обычно относятся эти системы, - проблема устранения состязаний - в данном случае не стоит достаточно остро, так как *ИМ* обладают большой инерционностью; а во-вторых, это свойство *ИМ* резко снижает требования к быстродействию схем. Указанные ограничения упрощают построение комбинационных схем в системах рассматриваемого класса, например, по сравнению с построением соответствующих схем, используемых в вычислительной технике.

1-2. СПОСОБЫ ЗАДАНИЯ АЛГОРИТМОВ РАБОТЫ УПРАВЛЯЮЩИХ ЛОГИЧЕСКИХ УСТРОЙСТВ

Не составляет секрета, что большинство реально созданных устройств логического управления построено непосредственно по словесному описанию алгоритмов их работы без использования формализованных методов. Последовательность проектирования их обычно такова: техническое задание в виде текста - формулы или функциональные схемы - принципиальные схемы. Формализованные методы, основанные на использовании автоматных таблиц, графов переходов и т. п., применяются лишь в некоторых наиболее сложных случаях, как правило, для построения и минимизации отдельных небольших подсхем. Объясняется это в значительной степени тем, что сложность описания при использовании упомянутых формализованных подходов определяется числом входов и состояний синтезируемого автомата и почти не зависит от сложности алгоритма и соответственно от сложности предстоящей схемной реализации. Так, размер таблицы состояний автомата, имеющего n входов и содержащего s триггеров, будет $2^n \times 2^s$, причем одинаковым как для самого простого, так и для самого сложного устройства.

Идеальным представляется случай, когда сложность описания алгоритма пропорциональна сложности его предстоящей схемной реализации. Забегая несколько вперед, отметим, что логические формулы обладают именно таким свойством: число букв в них линейно связано с числом элементов, требующихся для реализации этих формул. Но как написать формулу, минуя стадию составления таблиц? В ряде случаев это возможно непосредственно по словесному описанию. Однако иногда по словесному описанию записать формулу чрезвычайно трудно.

Из сказанного вытекают требования, которым должен отвечать формализованный язык для описания алгоритмов работы дискретных управляющих устройств: иметь структуру, близкую к структуре естественного языка; сложность описания на этом языке должна быть связана линейной зависимостью со сложностью предстоящей схемной реализации; допускать простой, однозначный и строго формализованный переход к логическим формулам.

Указанным требованиям в большей или меньшей степени отвечают так называемые импликационные языки, в основе которых лежит логическая конструкция «если x , то y ». Они продуктивно применяются в программировании, но при проектировании логических схем делаются лишь первые попытки их использования.

Ниже излагаются основные идеи, заложенные при разработке двух языков, ориентированных на решение задач логического синтеза: языка «секвенций» и языка «словных секвенций».

Прежде чем перейти к их изложению, остановимся на анализе первого из изложенных выше требований. Что имеется в виду под структурой, близкой к структуре естественного языка (словесного задания)? Чем естественный язык отличается от логической формулы? Основных отличий два: словесное описание позволяет задавать работу устройства (группы устройств) по частям, в то время как логическая формула требует указания всех условий срабатывания ИМ сразу; словесное описание ориентировано обычно на задание условий работы от входа к выходу, т. е. от причины к следствию, в то время как в логических формулах следствие предшествует причине.

Следовательно, формализованный язык, имеющий структуру, близкую к словесной, должен быть ориентирован на запись условий работы от причины к следствию и позволять записывать эти условия по частям. Последующий переход к логической формуле должен состоять в объединении этих частей по определенному правилу.

Секвенции. При использовании языка секвенций каждая составная часть сложного логического высказывания имеет вид «если $\varphi(x)=1$, то $\psi(y)=1$ », что записывается с помощью знака «секвенция» следующим образом:

$$\varphi(x) \vdash \psi(y) \quad (1-1)$$

Написанная выше секвенция означает, что при $\varphi(x)=1$ функция $\psi(y)$ также равна 1, но ничего не говорит о том, чему (0 или 1) равна $\psi(y)$ при $\varphi(x)=0$.

Составим функцию $z(\varphi; \psi)$, обращающуюся в единицу на наборах, не противоречащих секвенции $\varphi \vdash \psi$, и в нуль - на наборах, ей противоречащих (табл. 1-1).

Отметим, что ситуация, задаваемая третьей строкой таблицы, является единственной запрещенной выражением $\varphi \vdash \psi$ ситуацией.

По таблице нетрудно записать, что

$$z = \overline{\varphi} \vee \psi, \quad (1-2)$$

т. е. функция $z(\varphi; \psi)$ представляет собой импликацию [43].

Возможность сопоставления секвенций и импликаций облегчает выполнение математических преобразований над секвенциями, так как они сводятся к операциям, характерным для хорошо известного базиса $\{\&, \vee, \overline{}\}$.

Хотя в правой части секвенции возможна любая функция φ , в практике их использования ограничиваются случаем, когда ψ представляет собой конъюнкцию выходных переменных. Такие секвенции называют правильными.

Таблица 1-1
Таблица истинности

φ	ψ	z
0	0	1
0	1	1
1	0	0
1	1	1

Если одной секвенции соответствует импликация, то системе секвенций - произведение импликаций. Это утверждение следует из того факта, что две системы секвенций эквивалентны, т. е. описывают один и тот же алгоритм, если произведения соответствующих им импликаций равны. Частный, но весьма важный в практическом отношении случай эквивалентности устанавливается теоремой делимости: *системе правильных секвенций можно поставить в соответствие эквивалентную ей систему, у которой в правой части каждой секвенции содержится только одна переменная y_i из множества Y , а левая часть секвенции, соответствующей y_i , представляет собой дизъюнкцию левых частей тех исходных секвенций, в правые части которых входила y_i .* В разделенной системе знак секвенции можно заменить знаком равенства. Отсюда получается простое правило перехода от системы секвенций к системе логических формул: *для каждой выходной переменной надо сложить левые части тех секвенций, в которые она входит.*

Пример: Для системы секвенций вида

$$\begin{aligned} x_1 \vdash y_1 y_2 y_3; \\ \overline{x_2 x_3} \vdash y_1 y_2; \\ \overline{x_1 x_2} \vdash y_1 y_3; \\ x_4 \vdash y_2 \end{aligned}$$

определить соответствующую систему формул.

Пользуясь изложенным выше правилом, получим систему формул вида

$$\begin{aligned} y_1 &= x_1 \vee \overline{x_2 x_3} \vee \overline{x_1 x_2} = x_1 \vee \overline{x_2} \vee \overline{x_3}; \\ y_2 &= x_4 \vee \overline{x_2 x_3} = x_4 \vee \overline{x_2} \vee \overline{x_3}; \\ y_3 &= \overline{x_1 x_2} = \overline{x_1} \vee \overline{x_2}. \end{aligned}$$

Из теоремы делимости следует, что от системы секвенций возможен однозначный переход к логическим формулам в дизъюнктивной нормальной форме (ДНФ). При этом каждая секвенция является, как минимум, одним из членов ДНФ. Отсюда ясен выигрыш, который дает применение языка секвенций: вместо логических формул в целом мы можем задавать отдельные слагаемые ДНФ. Теорема делимости устанавливает, как эти частичные условия объединять в полные.

Задание программы работы в виде совокупности отдельных частей сопряжено с риском: не противоречат ли они друг другу? Противоречия возникают, если в правых частях присутствуют прямые и инверсные значения y_i . Признаком противоречия между условиями для y_i и $\overline{y_i}$ является то, что произведения левых частей соответствующих секвенций не обращаются в нуль.

Язык секвенций предусматривает наличие у *ИМ* памяти. Для того чтобы показать это, рассмотрим систему

$$\left. \begin{array}{l} f(x) \vdash y; \\ \varphi(x) \vdash \bar{y}. \end{array} \right\} \quad (1-3)$$

Предположим, что она непротиворечива. Это означает, что $f(x)\varphi(x) = 0$, т. е. если присутствует сигнал $\varphi(x)$, приводящий к $y = 0$, то сигнал $f(x)$, приведший к $y = 1$, в это время должен отсутствовать. Если бы при исчезновении сигнала $f(x)$ сигнал y обращался бы также в нуль, то необходимости в подаче сигнала $\varphi(x)$ не было бы. Отсюда следует, что при прекращении подачи $f(x)$ сигнал y сохраняет значение единицы, т. е. имеет место память. То, что язык секвенций предполагает наличие памяти у *ИМ*, ограничивает его применимость так называемыми автоматами с распределенной памятью. Необходимо отметить, что в рассматриваемом языке инверсия входной переменной означает сигнал на останов *ИМ*, а не сигнал запрета пуска *ИМ*. Если мы управляем совокупностью *ИМ*, часть из которых памятью не обладает, то существующая форма секвенциальной записи не содержит информации об этом, что, несомненно, является недостатком. Вместе с тем язык секвенций по своей конструкции во многом отвечает изложенным выше требованиям. Он, по мнению авторов, составляет хорошую основу для формализации логического синтеза, однако нуждается в усовершенствовании. Во-первых, желательно, чтобы язык записи алгоритмов был единым как для автоматов с распределенной памятью, так и для автоматов, *ИМ* которых памятью не обладают. Во-вторых, представляется полезным еще больше приблизить структуру языка к словесному описанию, т. е. задавать программу работы не в виде слагаемых ДНФ, а более мелкими частями. Наконец, следует предусмотреть правила, сводящие к минимуму возможную противоречивость в описании. Конечной целью в этом направлении является создание формализованного языка, позволяющего по словесному алгоритму, «под диктовку», записывать частичные логические условия и корректно объединять их в формулы. В определенной степени изложенным требованиям отвечает язык условных секвенций.

Условные секвенции. Условной секвенцией называется выражение вида

$$\varphi(x) \succ K(y) \quad (1-4)$$

где $\varphi(x)$ - некоторая функция от группы входных переменных; $K(y)$ - конъюнкция некоторой группы выходных переменных.

Это выражение означает, что если $\varphi(x) = 1$, то y_i , входящее в конъюнкцию $K(y)$ без отрицания, равно единице, если отсутствуют запрещающие условия.

Запрещающим условием для y_i является совокупность тех условных секвенций из числа образующих систему, в правые части которых y_i входит с отрицанием. При этом отметим, что в отличие от языка секвенций в данном случае инверсия выходной переменной означает не сигнал на останов, а запрещение пуска *ИМ*.

Очевидно, что система условных секвенций может содержать большое число условий. При этом никакая формализованная процедура не дает гарантии, что все условия приведены, и поэтому в качестве «расписки» заказчика в содержательной полноте системы следует записать оператор «Конец». Он свидетельствует о завершении задания условий работы системы. Далее, необходимо указать начальное состояние *ИМ*, выделив его оператором «Начало».

Приведем правило перехода от системы условных секвенций к системе логических формул:

- 1) для каждой выходной переменной, имеющей исходное состояние $y_i = 0$, сложить левые части тех условных секвенций, в которые она входит без отрицания;
- 2) полученную сумму умножить на произведение отрицаний левых частей тех секвенций, в которые y_i входит с отрицанием;
- 3) для выходных переменных с исходным состоянием $y_i = 1$ следует ограничиться вторым пунктом настоящего правила, т. е. записать y , как произведение отрицаний левых частей тех секвенций, в которые y_i входит с отрицанием.

Пример. Пусть задана следующая система условных секвенций. «Начало»:

$$y_1 = 0, y_2 = 0, y_3 = 1.$$

Программа:

$$\begin{aligned}x_1 &\succ y_1 y_2; \\x_2 x_3 &\succ \overline{y_1 y_2}; \\x_1 x_2 &\succ y_2 y_3; \\x_4 &\succ \overline{y_1}\end{aligned}$$

«Конец».

Тогда, руководствуясь изложенным правилом перехода, получим систему формул

$$\begin{aligned}y_1 &= \overline{x_1 x_2 x_3 x_4} = \overline{x_1 (x_2 \vee x_3) x_4}; \\y_2 &= \overline{(x_1 \vee \overline{x_1 x_2}) x_2 x_3} = \overline{(x_1 \vee x_2) (x_2 \vee x_3)}; \\y_3 &= \overline{x_1 x_2} = \overline{x_1 \vee x_2}.\end{aligned}$$

Язык условных секвенций позволяет описывать работу устройств, охваченных обратными связями.

Пример. Построить схему, имеющую два входа x_1 и x_2 и один выход y , работающую по следующей программе. При подаче сигнала x_1 на выходе появляется сигнал y , который запоминается, а при подаче сигнала x_2 этот сигнал исчезает.

Записывая словесный алгоритм на языке условных секвенций, получим:

«Начало»: $y = 0$.

Программа:

$$\begin{aligned}x_1 &\succ y; \\y &\succ y; \\x_2 &\succ \overline{y}\end{aligned}$$

«Конец». Отсюда

$$y = (x_1 \vee y) \overline{x_2};$$

Нетрудно видеть, что при составлении формулы приоритет был отдан сигналу x_2 , т. е. считалось, что

$$x_1 x_2 \succ \overline{y}.$$

Возникает вопрос, что делать, если мы желаем отдать предпочтение разрешающему сигналу, т. е. хотим, чтобы при одновременной подаче сигналов x_1 и x_2 выходной сигнал y был равен единице?

Для этой цели в языке условных секвенций предусмотрен оператор «НО». После оператора «НО» следует приводить те сочетания разрешающих и запрещающих переменных, для которых мы хотим сделать исключение с точки зрения приоритета.

Пример. Пусть задана следующая система условных секвенций.

«Начало»: $y = 0$.

Программа:

$$\begin{aligned}x_1 &\succ y; \\y &\succ y; \\x_2 &\succ \overline{y}; \\&\text{"НО"} \\x_1 x_2 &\succ y\end{aligned}$$

«Конец».

Как в этом случае перейти к булевой формуле? Для этого необходимо дополнить введенное ранее правило перехода от системы условных секвенций к системе логических формул следующим пунктом: 4) если в системе присутствует оператор «НО» то для каждой выходной переменной к формуле, полученной по условным секвенциям, записанным до оператора, прибавляются формулы, полученные по условным секвенциям, записанным после оператора.

Воспользовавшись этим правилом для рассмотренного выше примера, получим:

до оператора «НО» $y' = (x_1 \vee y)\bar{x}_2$;

после оператора «НО» $y'' = x_1x_2$.

Сумма

$y = y' \vee y'' = x_1\bar{x}_2 \vee yx_2 \vee x_1x_2 = x_1 \vee x_2y$.

Использование условных секвенций позволяет вести описание алгоритмов не только на микро-, но и на макроуровне. Было бы странным каждый раз заново строить триггеры, счетчики, дешифраторы и другие стандартные ячейки и блоки. Значительно удобнее рассматривать их как своеобразные исполнительные механизмы, одновременно являющиеся *ИИ* для последующих ступеней преобразований. Примеры использования условных секвенций на макроуровне приведены в работе [54].

В заключение отметим, что изложенные языки формализованного представления алгоритмов управляющих логических устройств нуждаются в дальнейшем совершенствовании и развитии, однако уже сегодня они позволяют достаточно просто переходить от словесного описания автомата к логическим формулам. Для перехода от логических формул к схемам также необходимы простые правила перехода, разработке которых, по существу, и посвящены следующие главы.

ГЛАВА ВТОРАЯ

ФУНКЦИОНАЛЬНЫЙ БАЗИС НАСТРАИВАЕМЫХ МОДУЛЕЙ

2-1. НАСТРАИВАЕМЫЕ МОДУЛИ. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Во введении мы отметили важность задачи создания ограниченной номенклатуры схемно-конструктивных единиц повышенного уровня интеграции, а также то, что наиболее сложной является разработка унифицированных модулей для построения комбинационных логических схем с нерегулярной структурой.

Действительно, умея строить, например, трехразрядный счетчик, мы без особого труда построим и двадцатиразрядный счетчик, так как регулярные схемы допускают простое наращивание. В то же время нерегулярные структуры с увеличением числа входов существенно изменяются и не могут рассматриваться как механическая сумма структур с меньшим числом входов.

Задача сокращения номенклатуры логических элементов обычно решается путем отказа от их схемной интеграции. При этом используется подход, основанный на известной теореме Поста - Яблонского о функциональной полноте [43]. Из нее следует, что любое логическое устройство можно построить с применением всего одного типа элементов «И - НЕ» либо «ИЛИ - НЕ».

Несмотря на то что такие элементы входят в состав большинства серий интегральных микросхем, выпускаемых промышленностью, их использование связано с рядом трудностей. Во-первых, проектант при описании алгоритмов работы использует обычно логические связки «И», «ИЛИ», «НЕ», вследствие чего построение функциональных схем обычно выполняется в базисе $\{\&, \vee, \bar{\quad}\}$, называемом булевым. При этом возникает необходимость их последующего перевода в принципиальные схемы из элементов «И - НЕ», «ИЛИ - НЕ». Процедура построения схем в этих базисах значительно более сложна, чем при использовании булева базиса. Это объясняется тем, что для операций «И - НЕ», «ИЛИ - НЕ» не выполняется сочетательный закон

$$x_1 | x_2 | x_3 \neq (x_1 | x_2) | x_3 \neq x_1 | (x_2 | x_3); \quad (2-1)$$

$$x_1 \downarrow x_2 \downarrow x_3 \neq (x_1 \downarrow x_2) \downarrow x_3 \neq x_1 \downarrow (x_2 \downarrow x_3), \quad (2-2)$$

в то время как этот закон имеет место для операции $\&$ и \vee :

$$x_1 \& x_2 \& x_3 = (x_1 \& x_2) \& x_3 = x_1 \& (x_2 \& x_3); \quad (2-3)$$

$$x_1 \vee x_2 \vee x_3 = (x_1 \vee x_2) \vee x_3 = x_1 \vee (x_2 \vee x_3); \quad (2-4)$$

Отметим, что этот закон выполняется и для операции «Сложение по модулю 2»:

$$x_1 \oplus x_2 \oplus x_3 = (x_1 \oplus x_2) \oplus x_3 = x_1 \oplus (x_2 \oplus x_3), \quad (2-5)$$

что обеспечивает удобство построения схем также и в базисе $\{\&, \vee, \bar{\quad}, \oplus\}$.

Логические элементы «И - НЕ» и «ИЛИ - НЕ» обладают еще и тем недостатком, что не удовлетворяют требованиям интегральной технологии - имеют большое число выводов, приходящихся на один вентиль. В результате в корпусе микросхемы может быть размещено лишь небольшое число вентилях, несмотря на то что интегральная технология позволяет реализовать сотни и тысячи вентилях на одном кристалле. Поэтому логические возможности интегральных микросхем (ИМС), содержащих элементы «И - НЕ», «ИЛИ - НЕ», относительно невысоки и требуется большое их число для реализации даже достаточно простых комбинационных схем.

Малый уровень интеграции таких элементов, кроме того, приводит к большому объему неунифицированного внешнего монтажа, что резко снижает надежность и увеличивает сложность печатных плат, используемых для размещения ИМС.

Удовлетворить требованиям интегральной технологии, упростить синтез, уменьшить объем внешнего монтажа, а также расширить функциональные возможности при небольшой номенклатуре элементов можно при использовании настраиваемых модулей.

Настраиваемым модулем называют устройство, которое при подаче на его входы внешних воздействий изменяет требуемым образом реализуемые им выходные функции.

При этом входы, связанные с источниками информации, от которых зависит рассматриваемая выходная функция, называют информационными входами модуля, а остальные входы - настроечными.

В настоящее время имеется большое число работ, в которых рассмотрены настраиваемые модули различных типов [9, 18, 19, 44, 58]. Для этих модулей может быть предложена классификация, проводимая по следующим основным признакам:

- а) по уровню функциональной и конструктивной интеграции: малого, среднего и большого уровня интеграции;
- б) по типу элементной базы: из функциональных элементов, из элементов с двусторонней проводимостью;
- в) по типу структуры: с однородной или с неоднородной структурой;
- г) по числу выходов: одновыходные, с прямым и инверсным выходами, многовыходные;
- д) по степени разделения информационных и настроечных входов: с полным разделением, с частичным разделением, с совмещением множества информационных и настроечных входов;
- е) по наличию памяти: автоматы с памятью или комбинационного типа;
- ж) по использованию памяти: модули, память в которых используется только для запоминания настройки; модули, в которых память используется как для запоминания настройки, так и в процессе работы;
- з) по способу введения информации: с параллельным вводом, с последовательным вводом, с параллельно-последовательным вводом;
- и) по типу настройки: с постоянной (необратимой) или с переменной настройкой;
- к) по способу переменной настройки, осуществляемой путем: подачи констант 0 и 1; отождествления входов; антиотождествления входов; выбора выхода многовыходного модуля; наложения перемычек на выходные выводы модуля; выбора модуля из набора;
- л) по виду объекта настройки: реализующие путем настройки автоматы; реализующие путем настройки булевы функции; настраиваемые на заданные конфигурации схем;
- м) по степени универсальности: универсальные или многофункциональные;
- н) по разновидностям реализуемых функций и схем: универсальные в классе всех булевых функций; универсальные в некотором классе булевых функций (в некотором классе схем); многофункциональные модули, реализующие путем настройки некоторый набор функций (схем).

В предыдущей главе уже отмечалось, что для построения управляющих логических устройств в основном должны использоваться модули комбинационного типа. Поэтому далее будут рассматриваться модули только этого типа, которые будем называть **настраиваемыми логическими модулями** (НЛМ).

С другой стороны, ввиду того что при построении логических устройств могут использоваться как функциональные элементы, так и элементы с двусторонней проводимостью, в книге будут рассматриваться методы построения и использования НЛМ на элементах каждой из указанных разновидностей.

2-2. ВЫБОР ФУНКЦИОНАЛЬНОГО БАЗИСА МОДУЛЕЙ

Функция, задающая однозначное соответствие между двоичными наборами входных переменных и двоичными значениями выходной переменной, называется булевой функцией.

Булевы функции задаются обычно двоичными таблицами, называемыми т а б л и ц а м и и с т и н н о с т и . При этом каждая из них характеризуется числом входных переменных n , от которого она зависит.

Аналитическая запись функций алгебры логики (ФАЛ), в которой используются символы булевых операций и переменных, называется булевой формулой. Одноместные и двухместные булевы операции, используемые при записи булевых формул, называются базисом этих формул.

Основными характеристиками булевой формулы являются ее базис и число символов входных переменных в ней. В дальнейшем символы входных переменных в булевой формуле будем называть буквами.

Если основной характеристикой ФАЛ является число переменных n , то для булевой формулы такой характеристикой является число букв h . Булева формула называется нормальной, если символы отрицаний расположены в ней лишь над одиночными символами переменных.

Отметим, что если задана некоторая булева формула из n букв, которая не принадлежит классу нормальных формул, то она может быть представлена эквивалентной (соответствующей той же функции) нормальной булевой формулой, содержащей то же число букв. В дальнейшем основное внимание будет уделяться нормальным булевым формулам, так как они используются обычно при описании алгоритмов работы управляющих логических устройств.

Класс функций (формул), реализуемых модулем путем настройки, называют функциональным базисом модуля.

Выбор функционального базиса определяет возможность выполнения ряда требований, которым должны удовлетворять разрабатываемые методы построения и использования НЛМ. Перечислим основные из них: а) возможность построения НЛМ, наиболее отвечающих потребностям класса управляющих логических устройств, для применения в которых они предназначены; б) приспособленность к решению задач большой размерности, т. е. к синтезу систем с многими входами и выходами; в) простота и удобство при «ручном» проектировании и возможность использования при автоматизации синтеза; г) предсказуемость результатов, т. е. гарантированное соответствие их некоторым оценкам сложности. Удовлетворить этим требованиям можно лишь при выборе функционального базиса НЛМ, учитывающем специфику булевых функций, описывающих алгоритмы функционирования управляющих логических устройств.

С целью рассмотрения особенностей этих функций были проанализированы более 2000 ФАЛ, описывающих нерегулярные логические схемы систем управления судовыми техническими средствами [54]. При этом было установлено, что булевы формулы, соответствующие этим функциям, неповторны либо обладают малой повторностью переменных в булевом базисе.

Напомним, что булева формула называется неповторной, если число букв в ней равно числу переменных. Для определения степени повторности переменных введем коэффициент Λ , вычисляемый как отношение числа букв в формуле h к числу независимых переменных в ней:

$$\Lambda = h/n \quad (2-6)$$

Бесповторность практически каждой булевой формулы ($\Lambda = 1$) в алгоритмах работы систем управления существует наряду с повторностью переменных в различных формулах одной системы, т. е. для устройств рассматриваемого класса характерна с в я з н о с т ь формул, количественной характеристикой которой является коэффициент связности, вычисляемый как отношение

суммарного числа букв $H = \sum_{i=1}^N h_i$ в системе из N формул к суммарному числу независимых

переменных $\prod = \sum_{i=1}^N n_i$ [54]. Анализ систем формул, соответствующих алгоритмам работы большого

числа устройств промышленной автоматики, показал, что средний коэффициент связности для каждой системы имеет небольшое значение и приблизительно равен 2 - 3.

Выводы относительно структуры булевых формул рассмотренного класса систем справедливы также и для других управляющих логических устройств. В работе [1] приведены формулы, описывающие системы управления газотурбинными установками компрессорных станций газопроводов. Рассмотрение формул, описывающих алгоритмы работы турбодетандера, стопорных и регулирующих клапанов,

маслонасосов и кранов, показывает, что каждая из них неповторна. Этот же вывод можно сделать из анализа формул, описывающих устройства управления оросительными системами [35] и различным технологическим оборудованием [36].

В работе [39] была проведена перепись наиболее распространенных схем автоматов, управляющих соединением в устройствах автоматической телефонии. Выбор этого класса устройств был обусловлен тем, что релейные устройства, используемые в автоматической телефонии, относятся к наиболее массовым и сложным релейно-контактным схемам. При этом было исследовано 1786 цепей, которые описывались 986 различными булевыми формулами. Выяснилось, что каждая из этих формул принадлежит

Таблица 2-1

Процентное соотношение разновидностей булевых функций в устройствах управления ЦВМ

Тип ЦВМ	Класс функций			
	А	Б/А	О/Б	Н/О
"Наири"	96,5	3,2	0,3	-
"Минск-2"	73,4	21,2	0,2	5,2
СЦВМ	71,5	22,0	2,8	3,7
БЭСМ-6	54	8	19	19

хотя бы одному из четырех классов упорядоченного типа: неповторным формулам; функциям, обладающим групповой инвариантностью; функциям, допускающим разделительную декомпозицию; однородным функциям.

Таким образом, и для этих устройств неповторность формул является характерным свойством, тем более что и однородные, и делимые функции обычно имеют малую повторность переменных в соответствующих формулах.

В настоящее время наиболее сложными устройствами логического управления являются устройства управления ЦВМ. Авторы работы [42] исследовали функции, описывающие эти устройства для различных ЦВМ. Результаты этой работы, выраженные в процентах, приведены в табл. 2-1, в которой А обозначает класс неповторных ДНФ; Б/А - класс неповторных скобочных формул; О/Б - класс однородных повторных функций и Н/О - класс неоднородных функций.

Таким образом, доля неповторных формул в устройствах управления указанных ЦВМ является основной и соответственно равна 99,7, 94,6, 93,5 и 62%.

Для выяснения влияния неупорядоченных функций на средний коэффициент повторности авторами были исследованы 155 представителей типов функций, на которые разбиваются 1704 функции, описывающие устройство управления ЦВМ «Минск-2». При этом оказалось, что лишь девять представителей типов не содержат неповторных формул, причем к ним относятся всего лишь 92 функции. В табл. 2-2 приведены указанные девять представителей и определены соответствующие коэффициенты повторности для каждого из них.

Средний элемент повторности для массива повторных формул

$$\wedge_{cp} = \frac{\sum_{i=1}^9 \wedge_i N_i}{\sum_{i=1}^9 N_i} = 1,10, \quad (2-7)$$

а средний коэффициент повторности для своего массива

$$\wedge_{cp} = \frac{\sum_{i=1}^{155} \wedge_i N_i}{\sum_{i=1}^{155} N_i} = 1,005. \quad (2-7)$$

Следовательно, и для рассмотренного класса устройств управления характерно, что описывающие их булевы формулы неповторны или близки к ним. При этом необходимо отметить, что многие формулы неповторны непосредственно в ДНФ. Так например из табл. 2-1 следует, что в устройствах управления ЦВМ «Наири» и «Минск-2» соответственно 96,5 и 73,4% всех формул неповторны в ДНФ.

**Представители типов повторных формул, описывающих
устройство управления ЦВМ «Минск-2»**

Представитель типа функций	Коэффициент повторности	Число функций, принадлежащих типу
$(x_1 \vee x_3)\bar{x}_2 \vee x_3$	1,67	1
$x_1(x_3 \vee x_4) \vee \bar{x}_2x_3$	1,25	1
$x_1\bar{x}_2 \vee x_4(\bar{x}_1 \vee \bar{x}_2 \vee x_3) \vee \bar{x}_5x_6$	1,34	1
$\bar{x}_1 \left[\bar{x}_4 (\bar{x}_2 \vee \bar{x}_3) \vee \bar{x}_2\bar{x}_3 \right] \vee x_4 \vee x_5x_6$	1,12	1
$\bar{x}_6[x_1(\bar{x}_2 \vee \bar{x}_3 \vee \bar{x}_4) \vee x_2\bar{x}_5] \vee x_7\bar{x}_8$	1,12	1
$x_1\bar{x}_2(x_5 \vee \bar{x}_6) \vee x_3x_4(x_5 \vee \bar{x}_7)$	1,14	1
$\bar{x}_3(x_1 \vee x_2) \vee \bar{x}_4 \vee \bar{x}_5 \vee \bar{x}_6x_7 \vee x_2x_8$	1,12	13
$\bar{x}_8[x_1(\bar{x}_2 \vee \bar{x}_3) \vee \bar{x}_3x_4 \vee x_7(\bar{x}_5 \vee \bar{x}_6)]$	1,12	1
$x_1\bar{x}_2 \vee x_4(x_3 \vee x_5) \vee x_6\bar{x}_7 \vee x_8x_9 \vee x_{10}\bar{x}_{11} \vee x_{11}x_{13}$	1,08	7

посредственно в ДНФ. Так, например, из табл. 2-1 следует, что в устройствах управления ЦВМ «Наири» и «Минск-2» соответственно 96,5 и 73,4% всех формул бесповторны в ДНФ.

Таким образом, из результатов проведенного исследования вытекает, что НЛМ для построения управляющих логических устройств, исходя из специфики булевых формул, описывающих алгоритмы их функционирования, должны быть способны реализовать путем настройки не произвольные функции n переменных, а лишь только те из них, для которых булевы формулы бесповторны или обладают малой повторностью переменных, что обеспечит построение модулей с малыми элементной сложностью и числом внешних выводов.

Наиболее вероятной причиной низкой повторности переменных в формулах, описывающих алгоритмы работы управляющих логических устройств, является использование принципа максимального воздействия при выборе источников информации (см. § 1-2). При таком выборе каждый ИИ, как правило, либо только способствует, либо только препятствует срабатыванию того или иного ИМ. Кроме того, оказывается возможным проследить влияние

Таблица 2-3

Характеристики схем, реализующих формулу

$y = (x_1x_2x_3 \vee x_4x_5)(x_6 \vee x_7)$ в различных базисах

Базис	Характеристики схем		
	Номенклатура элементов	Число элементов	Число внешних выводов
Двухвходные элементы «И» и «ИЛИ»	2	6	18
Трехвходовые элементы «И» и «ИЛИ»	2	5	20
Двухвходовые элементы «И-НЕ»	1	8	24
Трехвходовые элементы «И-НЕ»	1	6	24
Настраиваемые модули, универсальные в классе функций от трех переменных	1	3	18
Универсальный набор трехвходовых модулей	4(3)	3	12
Настраиваемые модули, универсальные в классе формул из трех букв	1	3	15

ИИ или их группы на состояние ИМ, т. е. произвести простую разделительную декомпозицию. При этом известно, что функциям, допускающим простую разделительную декомпозицию по всем переменным, соответствуют неповторные формулы.

Рассмотрев статистические характеристики булевых формул, определяющие выбор функционального базиса НЛМ, вернемся к вопросу о влиянии выбранного функционального базиса на такие характеристики схем из функциональных элементов (ФЭ), как номенклатура, количество элементов и суммарное число внешних выводов.

Указанный вопрос рассмотрим на примере формулы, неповторной в базисе $\{\&, \vee, \bar{}\}$, что, как было показано выше, характерно для рассматриваемого класса устройств.

Пусть требуется реализовать некоторую неповторную формулу в указанном базисе, например $y = (x_1 x_2 x_3 \vee x_4 x_5)(x_6 \vee x_7)$, на двухвходовых элементах «И» и «ИЛИ». На рис. 2-1, а приведена схема, построенная из этих элементов, которая характеризуется параметрами, приведенными в табл. 2-3.

Попытаемся уменьшить значения этих характеристик путем некоторого повышения уровня интеграции используемых элементов - применения трехвходовых элементов «И» и «ИЛИ» (рис. 2-1, б). Из табл. 2-3 следует, что при переходе к таким элементам номенклатура не изменилась, число элементов уменьшилось незначительно, а число внешних выводов возросло. Следовательно, повышение уровня интеграции элементов путем простого увеличения числа входов элементов «И» и «ИЛИ» не позволяет улучшить все указанные характеристики схемы.

Переход к элементам типа «И - НЕ» позволяет уменьшить номенклатуру составляющих схемы, однако их число и суммарное число внешних выводов при этом не уменьшаются (рис. 2-1, в, г).

Реализуем заданную формулу на существенно других элементах - настраиваемых логических модулях среднего уровня интеграции, универсальных в классе всех булевых функций от трех переменных (рис. 2-1, д). Использование таких модулей позволяет резко сократить число дискретных компонентов в схеме без увеличения их номенклатуры. Однако ввиду того, что каждый такой модуль имеет (при использовании в числе операций настройки операции «Антиотождествление») шесть внешних выводов (пять входов и один выход), то суммарное число внешних выводов компонентов схемы не уменьшается (табл. 2-3).

Таким образом, использование настраиваемых модулей, универсальных в классе всех булевых функций от трех переменных, также не позволяет улучшить все рассматриваемые характеристики схемы.

Суммарное число внешних выводов компонентов схемы может быть резко снижено при одновременном уменьшении числа этих компонентов, если в качестве базиса схемы использовать универсальный набор трехвходовых элементов, каждый из которых реализует одного из представителей типов формул в булевом базисе:

$$\varphi_1 = z_1 z_2 z_3; \quad \varphi_2 = z_1 \vee z_2 \vee z_3; \quad \varphi_3 = (z_1 \vee z_2) z_3; \quad \varphi_4 = z_1 z_2 \vee z_3.$$

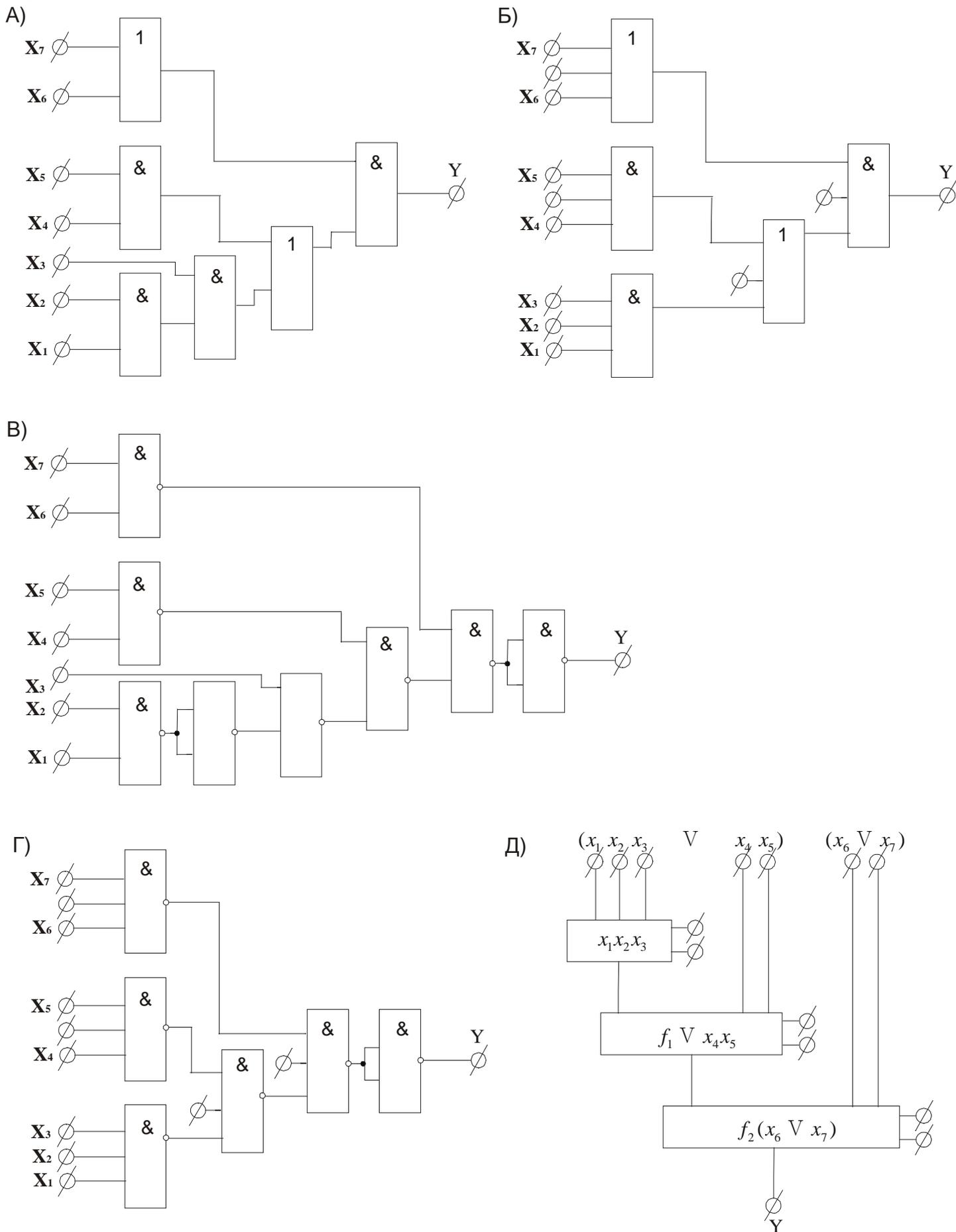
В этом случае (рис. 2-1, е) номенклатура универсального набора компонентов увеличивается до четырех (в данной схеме до трех), однако значения двух остальных характеристик резко уменьшаются (табл. 2-3). Поэтому, если удастся построить логический модуль, реализующий путем настройки каждого из указанных представителей типов формул из трех букв с пятью внешними выводами (четырьмя входами и одним выходом), то заданная формула будет реализована схемой (рис. 2-1, ж), обладающей наилучшими характеристиками из всех рассмотренных схем. Модуль, обладающий указанным числом внешних выводов, разработан на основе метода, изложенного в гл. 3, и выпускается в настоящее время в составе серии поликорпусных микросборок.

Приведенный пример характерен для неповторных формул и формул, обладающих малой повторностью переменных в булевом базисе. Для функций, реализуемых формулами с большой повторностью переменных, более эффективными могут оказаться модули, универсальные в классе всех логических функций [9]. Так, например, формула $y = (\bar{x}_1 x_2 \vee x_1 \bar{x}_2) x_3 \vee (\bar{x}_1 \bar{x}_2 \vee x_1 x_2) \bar{x}_3 = x_1 \oplus x_2 \oplus x_3$ требует для своей реализации всего лишь одного модуля, универсального в классе функций от трех переменных, с семью либо шестью внешними выводами, в то время как модуль, универсальный в классе булевых формул из 10 букв, или схема из модулей, универсальных в классе формул из меньшего числа букв, имеют значительно большее число внешних выводов.

Для реализации формул с большой повторностью переменных в булевом базисе, но обладающих малой повторностью при их представлении в расширенном базисе $\{\&, \vee, \bar{}, \oplus\}$, можно использовать

настраиваемые модули, универсальные в классе формул в базисе $\{\&, \vee, \bar{}, \oplus\}$, так как схемы, построенные из таких модулей, будут иметь меньшее число внешних выводов, чем схемы, выполненные на модулях, универсальных в классе всех булевых функций.

Таким образом, выбор в качестве функционального базиса НЛМ класса булевых формул либо класса формул $\{\&, \vee, \bar{}, \oplus\}$



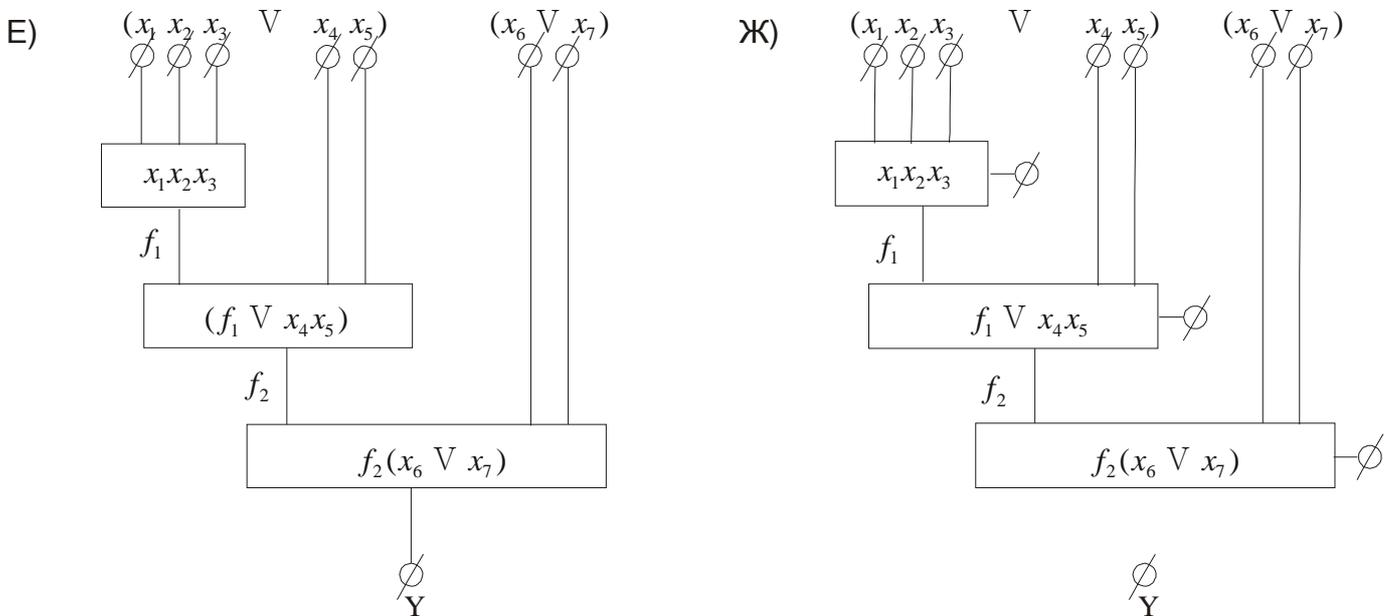


Рис. 2-1. Пример реализации неповторной булевой формулы на раз личных элементах: а - на двухвходовых элементах «И» и «ИЛИ»; б - на трехвходовых элементах «И» и ИЛИ»; в - на двухвходовых элементах «И - НЕ»; г - на трехвходовых элементах «И - НЕ»; д - на настраиваемых модулях, универсальных в классе функций от трех переменных; е - на универсальном наборе трехвходовых модулей; ж - на настраиваемых модулях, универсальных в классе формул из трех букв

обеспечивает сокращение аппаратных затрат при реализации формул, имеющих малую повторяемость переменных, по сравнению с затратами при использовании классических базисов, известных из литературы.

Необходимо отметить также, что для всех двухместных операций выбранного функционального базиса выполняется сочетательный закон. Это обеспечивает, как будет показано в гл. 4, возможность разработки простого метода рационального использования таких модулей и получения оценок сложности предстоящих схемных реализаций.

Модуль, реализующий путем настройки нормальные формулы длиной K_1 букв в базисе, для двухместных операций которого выполняется сочетательный закон, будем называть K_1 - многофункциональным. Если многофункциональный модуль реализует все нормальные формулы указанного базиса из K букв, то он является K - универсальным.

Универсальность модуля в классе неповторных формул обеспечивает также его универсальность и в классе произвольных нормальных формул из того же числа букв. Базис, разумеется, предполагается одинаковым.

Например, модуль, настроенный на неповторную формулу $x_1x_2 \vee x_3x_4$, может реализовать также и повторную формулу $\overline{z_1}z_2 \vee \overline{z_1}z_2$. Для этого надо на входы x_1 и x_4 подать соответственно переменные z_1 и z_2 , а на входы x_2 и x_3 - переменные $\overline{z_2}$ и $\overline{z_1}$.

Реализация повторных формул из K букв в базисе K -универсальных модулей осуществляется в два этапа: а) на первом этапе модуль настраивается на реализацию неповторной формулы, структура которой совпадает со структурой заданной формулы; б) на втором этапе у модуля, настроенного указанным способом, осуществляется отождествление входов.

Таким образом, НЛМ, первоначально выбранные на основе исследования специфики схем, описываемых неповторными формулами, пригодны также и для реализации любых других формул, хотя при этом эффективность их использования снижается, так как для подачи одной переменной приходится использовать несколько выводов, как в приведенном выше примере.

Возникает вопрос: если НЛМ должны выполнять неповторные формулы от K , букв, то сколько этих формул существует и каковы они? Иными словами, в первую очередь необходимо рассмотреть вопросы подсчета, классификации и табулирования неповторных формул в выбранных базисах $\{\&, \vee, \overline{\quad}\}$ и $\{\&, \vee, \overline{\quad}, \oplus\}$, чему и посвящен следующий параграф.

2-3. СВОЙСТВА БЕСПОВТОРНЫХ БУЛЕВЫХ ФОРМУЛ

Число PN -типов неповторных формул в базе {&, ∨, ¬}.

Важнейшей работой, посвященной этому вопросу, является статья Дж. Риордана и К. Шеннона «Число двухполюсных параллельно-последовательных схем» [57]. В ней рассматривались

Таблица 2-4

Число представителей неповторных двухполюсных П – схем

Число схем	Число контактов d									
	1	2	3	4	5	6	7	8	9	10
$r(d)$	1	2	4	9	22	57	154	429	1225	3565
$S_{PN}(d)$	1	2	4	10	24	66	180	522	1532	4624
$t(d)$	1	2	4	10	28	84	264	858	2860	9724

Неизоморфные двухполюсные неповторные параллельно-последовательные схемы и определялось их число.

В табл. 2-4 приведены значения числа представителей рассматриваемых схем $S_{PN}(d)$ при $d = 1 \div 10$, где d – число контактов в схеме.

В указанной работе найдены также рекуррентные соотношения для определения нижних и верхних оценок числа этих схем:

$$r(d) = \frac{3}{2}r(d-1) + \frac{1}{4} \sum_{i=1}^{d-1} r(d-i)r(i), \quad (2-9)$$

$$t(d) = t(d-1) + \frac{1}{2} \sum_{i=1}^{d-1} t(d-i)t(i), \quad (2-10)$$

где

$$r(1) = t(1) = 1, \quad r(2) = t(2) = 2.$$

При этом

$$t(d) \leq S_{PN}(d) \leq r(d). \quad (2-11)$$

Асимптотическая оценка числа схем $S_{PN}(d)$ имеет вид

$$S_{PN}(d) \approx \frac{3}{7}(3,56)^d * d^{-3/2}. \quad (2-12)$$

Наличие изоморфизма между П-схемами и булевыми формулами приводит к тому, что все результаты, полученные в работе [57], могут быть перенесены на класс неповторных в булевом базисе формул. Однако в силу того, что в рассмотренной работе авторы не различают замыкающие и размыкающие контакты, а также схемы, эквивалентные относительно перестановок их последовательных и параллельных частей, можно утверждать, что эти результаты применимы лишь для определения числа PN - типов неповторных формул.

Булева формула из h букв $f(x_1, \dots, x_h)$ принадлежит одному типу с формулой из h букв $g(x_1, \dots, x_h)$, если $f(x_1, \dots, x_h)$ переходит в $g(x_1, \dots, x_h)$ при некоторой перестановке переменных и замене некоторых переменных их отрицаниями.

Классификация булевых формул по типам относительно указанных операций называется PN - классификацией (от английских слов permutation - перестановка и negation - отрицание). Практически однотипность по PN - классификации соответствует возможности подачи на входы схемы произвольного сочетания входных переменных и их отрицаний.

Примеры однотипных формул:

$$x_1(x_2 \vee \bar{x}_3) \text{ и } x_4(\bar{x}_2 \vee \bar{x}_1);$$

$$x_1x_2 \vee x_3x_4 \text{ и } \bar{x}_1x_3 \vee \bar{x}_1x_3 \vee \bar{x}_2\bar{x}_4.$$

В результате классификации множество булевых формул распадается на попарно-непересекающиеся классы - множества однотипных формул. Каждую формулу данного класса можно выбрать в качестве представителя этого класса. Булевы формулы, принадлежащие одному классу, реализуются физически одинаковыми схемами. Поэтому для каждого класса достаточно реализовать

лишь одну схему, структура которой описывается формулой представителя класса. Получение любой формулы, принадлежащей классу, при этом осуществляется путем перестановки переменных на входе схемы-представителя за счет изменения подключения входов и подачи на входы прямых и инверсных значений входных переменных. При неравной доступности выходов *III* инверсные значения входных переменных получают от инверторов, располагаемых обычно независимо от модулей, например, на отдельной плате. В случае, если инверсные значения входных переменных недоступны, а использование отдельных инверторов нежелательно, то переходят к *P* - классификации, которая будет рассмотрена ниже.

Использование введенной классификации позволяет объединять между собой лишь представителей типов заданных функций, а не сами функции, что резко упрощает процесс построения модулей. Знание числа представителей типов функций недостаточно для построения НЛМ. Для этой цели необходимо перечисление всех представителей, которые должны быть объединены в порождающую функцию модуля. Поэтому рассмотрим вопрос о табулировании представителей указанного класса формул.

Табулирование представителей *PN*-типов бесповторных формул в базисе $\{\&, \vee, \bar{}\}$. Рассмотрим процедуру образования бесповторных формул, несколько отличную от изложенной в работе [57]. Этот подход более приспособлен для целей табулирования. В его основе лежит определение количества разбиений

Таблица 2-5

Число разбиений числа *h* на слагаемые

<i>h</i>	1	2	3	4	5	6	7	8
<i>R(h)</i>	1	2	3	5	7	11	15	22

R(h) числа *h* на сумму целых ненулевых слагаемых. В работе [49] приведено рекуррентное соотношение для определения числа разбиений, которое имеет следующий вид:

$$R(h) = \sum_{k=1}^{h-1} (-1)^{k-1} \left[R\left(h - \frac{3k^2 - k}{2}\right) + R\left(h - \frac{3k^2 + k}{2}\right) \right]. \quad (2-13)$$

Все аргументы в правой части этого соотношения должны быть неотрицательными, а граничное условие должно иметь значение, равное единице: $R(0) = 1$. В табл. 2-5 приведены значения *R(h)* при $h = 1 \div 8$.

Соотношение (2-13) может быть использовано также для подсчета числа представителей *PN*-типов бесповторных ДНФ из *h* букв, так как существует взаимно-однозначное соответствие между представителями *PN*-типов бесповторных ДНФ и разбиениями.

Например, при $h = 3$ существуют три *PN*-типа бесповторных ДНФ, представители которых имеют следующий вид: $f_1 = x_1 x_2 x_3$, $f_2 = x_1 x_2 \vee x_3$, $f_3 = x_1 \vee x_2 \vee x_3$. Им соответствуют следующие разбиения: 3; 2+1; 1+1+1.

Воспользуемся $0(h) - R(h) - 1$ разбиениями (всеми разбиениями за исключением собственно числа *h*) для образования представителей *PN* - типов бесповторных в базисе $\{\&, \vee, \bar{}\}$ формул, в которых последней операцией является дизъюнкция (формулы первого вида). Так как среди бесповторных формул отсутствуют самодвойственные [43], то каждой бесповторной формуле, в которой последней операцией является дизъюнкция, соответствует двойственная ей формула, в которой последней операцией является конъюнкция (формулы второго вида).

Ввиду того что в бесповторных формулах все буквы различны или имеют различные индексы, с целью более компактной их записи введем представление *PN* - типов этих формул в виде арифметических полиномов.

Таблица 2-6

Представители PN -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$

h	$O(h)$	№ п/п	Разбиения	Формулы первого вида	$S_{\vee}(h)$	Формулы второго вида	$S_{\&}(h)$	$S_{PN}(h)$
2	1	1	1+1	1+1	1	2	1	2
3	2	1	2+1	2+1	1	(1+1)1	1	4
		2	1+1+1	1+1+1	1	3	1	
4	4	1	3+1	(1+1)1+1 3+1	2	(2+1)1 (1+1+1)1	2	10
		2	2+2	2+2	1	(1+1)(1+1)	1	
		3	2+1+1	2+1+1	1	(1+1)2	1	
		4	1+1+1+1	1+1+1+1	1	4	1	
5	6	1	4+1	(2+1)1+1 (1+1+1)1+1 (1+1)(1+1)+1 (1+1)2+1 4+1	5	[(1+1)1+1]1 (3+1)1 (2+2)1 (2+1+1)1 (1+1+1+1)1	5	24
		2	3+1	(1+1)1+2 3+2	2	(2+1)(1+1) (1+1+1)(1+1)	2	
		3	3+1+1	(1+1)1+1+1 3+1+1	2	(2+1)2 (1+1+1)2	2	
		4	2+2+1	2+2+1	1	(1+1)(1+1)1	1	
		5	2+1+1+1	2+1+1+1	1	(1+1)3	1	
		6	1+1+1+1+1	1+1+1+1+1	1	5	1	

Таблица 2-7

Число PN -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$

h	$O(h)$	№ п/п	Разбиения	$S_{\vee}(h)$	$S_{\&}(h)$	$S_{PN}(h)$
6	10	1	5+1	12	12	66
		2	4+2	5	5	
		3	4+1+1	5	5	
		4	3+3	3	3	
		5	3+2+1	2	2	
		6	3+1+1+1	2	2	
		7	2+2+2	1	1	
		8	2+2+1+1	1	1	
		9	2+1+1+1+1	1	1	
		10	1+1+1+1+1+1	1	1	

7	14	1	6+1	33	33	180
		2	5+2	12	12	
		3	5+1+1	12	12	
		4	4+3	10	10	
		5	4+2+1	5	5	
		6	4+1+1+1	5	5	
		7	3+3+1	3	3	
		8	3+2+2	2	2	
		9	3+2+1+1	2	2	
		10	3+1+1+1+1	2	2	
		11	2+2+2+1	1	1	
		12	2+2+1+1+1	1	1	
		13	2+1+1+1+1+1	1	1	
		14	1+1+1+1+1+1+1	1	1	
8	21	1	7+1	90	90	522
		2	6+2	33	33	
		3	6+1+1	33	33	
		4	5+3	24	24	
		5	5+2+1	12	12	
		6	5+1+1+1	12	12	
		7	4+4	15	15	
		8	4+3+1	10	10	
		9	4+2+2	5	5	
		10	4+2+1+1	5	5	
		11	4+1+1+1+1	5	5	
		12	3+3+2	3	3	
		13	3+3+1+1	3	3	
		14	3+2+2+1	2	2	
		15	3+2+1+1+1	2	2	
		16	3+1+1+1+1+1	2	2	
		17	2+2+2+2	1	1	
		18	2+2+2+1+1	1	1	
		19	2+2+1+1+1+1	1	1	
		20	2+1+1+1+1+1+1	1	1	
		21	1+1+1+1+1+1+1+1	1	1	

Арифметическим полиномом будем называть символическую запись булевой формулы посредством арифметического выражения, в котором числа изображают ранг каждой конъюнкции, а знаки суммы и произведения соответствуют операциям дизъюнкции и конъюнкции в формуле.

Обезличенное представление структуры формулы в виде арифметического полинома во многом аналогично представлению структуры контактной схемы ее «скелетом» [13]. Представление формул в виде арифметических полиномов имеет смысл не только при классификации неповторных формул, но и при допуски мости переобозначения входных переменных, и для классификации произвольных формул. При этом арифметические полиномы могут использоваться как представители типов таких формул. Поэтому, подсчитывая число представителей PN -типов неповторных нормальных формул, мы тем самым определяем и число представителей типов произвольных нормальных формул, заданных в том же базисе.

Введение представления формул в виде арифметических полиномов позволяет использовать разбиения для табулирования представителей рассматриваемого класса формул. При этом построение формул первого вида осуществляется путем подстановки в разбиения вместо каждой из цифр всех формул второго вида из меньшего числа букв. В табл. 2-6 приведен процесс образования представителей PN -типов неповторных формул в базисе $\{\&, \vee, _ \}$ при $h = 2 \div 5$.

Формулы, содержащие большее число букв, строятся по тому же правилу, однако при этих значениях L необходимо учитывать то, что некоторые разбиения порождают изоморфные формулы, из которых в качестве представителя типа должна выбираться одна из них.

Таблица 2-8

Число двухполюсных П-схем с различными ярлыками

Число схем	Число букв h									
	1	2	3	4	5	6	7	8	9	10
$S_N(h)$	1	2	8	52	472	5504	78416	1320064	25637824	564275648

Появление изоморфных формул связано с наличием в некоторых разбиениях одинаковых цифр (при $h \geq 6$). При $h = 6$ существует лишь одно разбиение, порождающее изоморфные формулы.

Действительно, подставляя в разбиение $3 + 3$ формулы второго вида из трех букв, получим четыре формулы: $(1+1)1+3$; $3+(1+1)1$; $(1+1)1+(1+1)1$; $3+3$, из которых только три являются различными, а первые две совпадают. При $h = 7$ также существует лишь одно такое разбиение $3+3+1$, а при $h = 8$ - уже три разбиения, порождающих изоморфные формулы: $4+4$; $3 + 3+2$; $3 + 3+1$.

В табл. 2-7 приведены разбиения для $h = 6 \div 8$ и указано число типов формул, получаемых из каждого разбиения.

На основе подхода, изложенного выше, авторами были протабулированы все представители PN -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из $h = 1 \div 8$ букв.

При этом все представители типов формул из h букв были разбиты на группы, каждая из которых может быть реализована одной конфигурацией древовидной схемы, а ее элементы могут выполнять одну из операций базиса $\{\&, \vee, \bar{\quad}\}$.

Число N -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$. В книге Дж. Риордана [49]

приводятся производящие функции для определения числа параллельно-последовательных схем, снабженных различными ярлыками (различными индексами в обозначениях контактов). При этом схемы, отличающиеся лишь наличием замыкающих контактов вместо размыкающих и наоборот, считаются различными. Численные значения различных схем для этого случая приведены в табл. 2-8.

Этим схемам соответствует подкласс формул, неповторных в базисе $\{\&, \vee, \bar{\quad}\}$, в котором представители типов могут отличаться не только структурой, но и перестановкой переменных. Этот подкласс формул формируется при использовании N -классификации. При этом формулы, отличающиеся только расстановкой инверсий, считаются неразличимыми и принадлежат одному N -классу.

Число формул этого класса, порождаемых одним PN -типом неповторных формул из h букв,

$$S_N(h) = \frac{h!}{\prod (m_i!)}, \quad (2-14)$$

где m_i - число букв в конъюнкции или число одинаковых членов в дизъюнкции.

Пример. Представитель PN -типа $f \equiv (1+1)1+1$ порождает

$$S_N(h) = \frac{4!}{2!} = 12$$

различных N -типов неповторных формул:

$$\begin{array}{lll} f_1 = (x_1 \vee x_2)x_3 \vee x_4; & f_5 = (x_1 \vee x_4)x_2 \vee x_3; & f_9 = (x_2 \vee x_4)x_1 \vee x_3; \\ f_2 = (x_1 \vee x_2)x_4 \vee x_3; & f_6 = (x_1 \vee x_4)x_3 \vee x_2; & f_{10} = (x_2 \vee x_4)x_3 \vee x_1; \\ f_3 = (x_1 \vee x_3)x_2 \vee x_4; & f_7 = (x_2 \vee x_3)x_1 \vee x_4; & f_{11} = (x_3 \vee x_4)x_1 \vee x_2; \\ f_4 = (x_1 \vee x_3)x_4 \vee x_2; & f_8 = (x_2 \vee x_3)x_4 \vee x_1; & f_{12} = (x_3 \vee x_4)x_2 \vee x_1; \end{array}$$

Число неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$. Значение числа N – типов неповторных формул $S_N(h)$ позволяет определить

Таблица 2-9

Число формул	Число букв h									
	1	2	3	4	5	6	7	8	9	10
$S(h)$	2	8	64	832	15104	352256	10037248	338×10^6	13×10^9	578×10^9

Число неповторных формул $S(h)$ в базисе $\{\&, \vee, \bar{\quad}\}$. Несмотря на то, что $S(h)$ вычисляется по $S_N(h)$ достаточно просто:

$$S(h) = S_N(h)2^h, \quad (2-15)$$

в литературе до настоящего времени число неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ не приводилось.

В табл. 2-9 указаны значения $S(h)$ при $h = 1 \div 10$.

Пример. Перечислим все неповторные в рассматриваемом базисе формулы из двух букв:

$$\begin{array}{llll} f_1 = x_1 x_2; & f_3 = x_1 \bar{x}_2; & f_5 = x_1 \vee x_2; & f_7 = x_1 \vee \bar{x}_2; \\ f_2 = \bar{x}_1 x_2; & f_4 = \bar{x}_1 \bar{x}_2; & f_6 = \bar{x}_1 \vee x_2; & f_8 = \bar{x}_1 \vee \bar{x}_2. \end{array}$$

Для характеристики класса неповторных формул авторами введен коэффициент

$$B(h) = \frac{\log_2 S(h)}{\log_2 S_{PN}(h)}. \quad (2-16)$$

Значения этого коэффициента при $h = 2 \div 9$ приведены в табл. 2-10.

Таблица 2-10

h	2	3	4	5	6	7	8	9
$B(h)$	3	3	2,92	2,98	3,05	3	3,02	3,09

Таким образом, можно утверждать, что при рассмотренных малых значениях числа букв $B(h) \approx 3$. Эта величина может использоваться в качестве инварианта класса неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$.

Число P - типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$. Представители P - типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ образуют класс формул, в котором формулы, отличающиеся лишь перестановкой переменных, неразличимы. При P - классификации различают лишь формулы, отличающиеся структурой и расстановкой инверсий над одиночными символами переменных.

Путем непосредственного подсчета были найдены для каждого PN -типа неповторных формул порождаемые им представители P - типов формул, что позволило определить их суммарное число при $h = 1 \div 6$ (табл. 2-11).

Таблица 2-11

Число P - типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$

h	1	2	3	4	5	6
$S_P(h)$	2	6	20	80	340	1582

Пример. Существует шесть P - типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из двух букв, представителями которых являются: $f_1 = x_1 x_2$; $f_2 = \bar{x}_1 x_2$; $f_3 = x_1 \bar{x}_2$; $f_4 = x_1 \vee x_2$; $f_5 = \bar{x}_1 \vee x_2$; $f_6 = \bar{x}_1 \vee \bar{x}_2$.

Перечислим также представителей P - типов формул из трех букв:

$$\begin{array}{llll}
f_5 = \overline{(x_1 \vee x_2)}x_3; & f_{11} = \overline{x_1 x_2} \vee x_3; & & \\
f_1 = \overline{x_1 x_2 x_3}; & f_6 = \overline{(x_1 \vee x_2)}x_3; & f_{12} = \overline{x_1 x_2} \vee x_3; & f_{17} = \overline{x_1} \vee \overline{x_2} \vee \overline{x_3}; \\
f_2 = x_1 x_2 x_3; & f_7 = (x_1 \vee x_2)x_3; & f_{13} = \overline{x_1 x_2} \vee x_3; & f_{18} = \overline{x_1} \vee \overline{x_2} \vee x_3; \\
f_3 = \overline{x_1 x_2} x_3; & f_8 = (x_1 \vee x_2)x_3; & f_{14} = \overline{x_1 x_2} \vee x_3; & f_{19} = \overline{x_1} \vee x_2 \vee x_3; \\
f_4 = x_1 x_2 x_3; & f_9 = (x_1 \vee x_2)\overline{x_3}; & f_{15} = x_1 x_2 \vee \overline{x_3}; & f_{20} = x_1 \vee x_2 \vee x_3; \\
& f_{10} = (x_1 \vee x_2)x_3; & f_{16} = x_1 x_2 \vee x_3; &
\end{array}$$

Число NPN – типов неповторных формул в базисе $\{\&, \vee, \overline{}\}$.

Рассмотрев PN – классификацию, а также классификации с большим числом представителей, остановимся на классификации, позволяющей сократить число этих представителей.

При применении NPN – классификации используются три операции: замена формулы её отрицанием N , перестановка переменных P и замена переменных их отрицаниями N [7]. Введение дополнительной операции позволяет сократить число представителей типов формул по сравнению с их числом в PN – классификации в два раза (табл. 2-12).

Некоторые свойства неповторных формул. Задача построения настраиваемых логических модулей, выполняющих любую

Таблица 2-12

Число NPN – типов неповторных формул в базисе $\{\&, \vee, \overline{}\}$

h	1	2	3	4	5	6	7	8
$S_{NPN}(h)$	1	1	2	5	12	33	90	261

формулу длиной K букв, требует для своего решения знания некоторых свойств неповторных формул и соответствующих им функций. Для этого изложим несколько вопросов, которые могут показаться абстрактными, но на самом деле пригодятся в дальнейшем. Они относятся к числу единиц и структуре их расположения в таблицах истинности, соответствующих неповторным формулам.

Число единиц в столбце значений булевой функции условимся называть рангом. Если имеются формулы, зависящие каждая от своего множества переменных:

$f_1(x_1, \dots, x_{h_1}), f_2(x_{h_1+1}, \dots, x_{h_1+h_2}), \dots, f_k(x_{h_1+\dots+h_{k-1}}, \dots, x_{h_1+\dots+h_{k-1}+h_k})$ - и имеющие ранги соответственно r_1, r_2, \dots, r_i , то ранг их произведения $F = f_1 f_2 \dots f_i$ равен произведению рангов $r = r_1 r_2 \dots r_i$.

В качестве примера рассмотрим произведение двух функций f_1 и f_2 .

Ясно, что на тех наборах переменных, где $f_1 = 0$, и произведение $f_1 f_2 = 0$, а также на тех наборах переменных, где $f_1 = 1$, функция $F = f_2$.

Таким образом, если в таблице истинности сначала разместить переменные от f_1 , а потом – от f_2 , то столбец значений $F = f_1 f_2$ будет состоять из 2^{h_1} фрагментов длиной 2^{h_2} двух разновидностей: состоящих сплошь из 0 на наборах x_1, \dots, x_{h_1} , где $f_1 = 0$, и повторяющихся f_2 на наборах x_1, \dots, x_{h_1} , где $f_1 = 1$.

Пример. Определить ранг функции $F_1 = f_1 f_2$, если $f_1 = x_1 \oplus x_2$, а $f_2 = \overline{x} \vee x_4$.

Составим таблицы истинности для функций f_1 и f_2 (табл. 2-13, 2-14).

Ввиду того, что $r_1 = 2$, а $r_2 = 3$, на основании высказанного утверждения $r = r_1 r_2 = 6$. Проверим полученный результат (табл. 2-15).

Таблица 2-13
Таблица истинности для f_1

x_1	x_2	f_1
0	0	1
0	1	0
1	0	0
1	1	1

Таблица 2-14
Таблица истинности для f_2

x_1	x_2	f_1
0	0	1
0	1	0
1	0	0
1	1	1

Таблица 2-15

Таблица истинности для F_1 и F_2

x_1	x_2	x_3	x_4	f_1	f_2	$F_1 = f_1 f_2$	$F_2 = f_1 \vee f_2$
0	0	0	0	1	1	1	1
0	0	0	1	1	1	1	1
0	0	1	0	1	0	0	1
0	0	1	1	1	1	1	1
0	1	0	0	0	0	0	1
0	1	0	1	0	1	0	1
0	1	1	0	0	0	0	0
0	1	1	1	0	1	0	1
1	0	0	0	0	1	0	1
1	0	0	1	0	1	0	1
1	0	1	0	0	0	0	0
1	0	1	1	0	1	0	1
1	1	0	0	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	1	1	1

Ранг суммы двух формул вычисляется несколько сложнее, в соответствии с выражением

$$r = r_2 \times 2^{h_1} + r_1(2^{h_2} - r_2). \quad (2-17)$$

Для формул, рассмотренных в предыдущем примере, ранг суммы $F_2 = f_1 \vee f_2$ составит

$$r = 3 \times 2^2 + 2(2^2 - 3) = 14.$$

Это легко проверить, выполнив сложение функций f_1 и f_2 (табл. 2-15).

Отметим интересное следствие. Пусть ранг формулы $f_1(x_{n-1}, \dots, x_1)$, неповторной в базисе $\{\&, \vee, \bar{\quad}\}$, равен r_1 ; тогда формула $f_2(x_n, x_{n-1}, \dots, x_1) = x_n \vee f_1(x_{n-1}, \dots, x_1)$ имеет ранг $r_2 = 2^{h-1} + r_1$.

Полученные соотношения позволяют вычислять ранг любой неповторной формулы в указанном базисе аналитически, без построения истинности.

Пример. Определить ранг формулы $F = \{[(x_1 \vee x_2)x_3 \vee x_4]x_5 \vee x_6\}x_7$.

$$\begin{aligned} f_1 &= x_1; & r_1 &= 1; \\ f_2 &= x_1 \vee x_2; & r_2 &= 2^1 + 1 = 3; \\ f_3 &= (x_1 \vee x_2)x_3; & r_3 &= 3; \\ f_4 &= (x_1 \vee x_2)x_3 \vee x_4; & r_4 &= 2^3 + 3 = 11; \\ f_5 &= [(x_1 \vee x_2)x_3 \vee x_4]x_5; & r_5 &= 11; \\ f_6 &= [(x_1 \vee x_2)x_3 \vee x_4]x_5 \vee x_6; & r_6 &= 2^5 + 11 = 43; \\ f_7 &= \{[(x_1 \vee x_2)x_3 \vee x_4]x_5 \vee x_6\}x_7; & r_7 &= 43. \end{aligned}$$

Ранг функции, несущественно зависящей от некоторых своих переменных, есть четное число, так как если в функции $f(x_n, \dots, x_1)$ имеется хотя бы одна несущественная переменная, например с порядковым номером n , то столбец значений этой функции может быть построен удвоением столбца значений функции $f(x_{n-1}, \dots, x_1)$. Следовательно, вне зависимости от ранга функции $f(x_{n-1}, \dots, x_1)$ ранг функции $f(x_n, \dots, x_1)$ четен.

Приведем еще ряд утверждений. Если булева функция имеет, нечетный ранг, то она существенно зависит от всех своих переменных [56]. Ранг формулы, неповторной в базисе $\{\&, \vee, \bar{}\}$, существенно зависящей от всех своих переменных, нечетен. Инвариантами PN -типов неповторных пороговых функций, существенно зависящих от n переменных, являются их ранги (нечетные числа): $1, 3, 5, \dots, 2^n - 1$.

Классификация и табулирование неповторных формул в базисе $\{\&, \vee, \bar{}, \oplus\}$. В работе [66] приведено соотношение для определения числа PN -типов неповторных формул в базисе $\{\&, \vee, \bar{}, \oplus\}$, реализуемых линейной цепочкой из элементов, каждый из которых после настройки выполняет одну из трех двухместных операций $\{\&, \vee, \oplus\}$ над информационными переменными.

Указанное соотношение имеет вид

$$B_1(h) = \frac{1}{\sqrt{5}} \left[\left(\frac{3+\sqrt{5}}{2} \right)^h - \left(\frac{3-\sqrt{5}}{2} \right)^h \right] \approx \frac{(2,62)^h}{2,24}. \quad (2-18)$$

Интересным является тот факт, что значения $B_1(h)$, вычисленные по этой формуле, равны целым числам, являющимся нечетными членами ряда Фибоначчи:

$$B_1(h) = \Phi(2h+1). \quad (2-19)$$

Числа Фибоначчи определяются из соотношения

$$\Phi(h) = \Phi(h-1) + \Phi(h-2), \quad (2-20)$$

где $\Phi(1)=0, \Phi(2)=1$.

Ряд Фибоначчи имеет вид: $0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377$ и т.д.

В табл. 2-16 приведены значения $B_1(h)$ при $h = 1 \div 10$.

Отметим, что при $h \geq 3$

$$B_1(h) < 3^{h-1}. \quad (2-21)$$

Это связано с явлением изоморфизма, вызываемым операцией \oplus . Поэтому в базисе $\{\&, \vee, \bar{}, \oplus\}$ формулы, обладающие разной структурой, могут принадлежать одному PN -типу.

Таблица 2-16

Число PN -типов неповторных формул в базисе $\{\&, \vee, \bar{}, \oplus\}$

h	1	2	3	4	5	6	7	8	9	10
$B_1(h)$	1	3	8	21	55	144	377	987	2584	6765
$B_2(h)$	1	3	8	29	101	405	?	?	?	?
$B_3(h)$	1	3	8	42	165	864	4147	22701	118864	662970

Таблица 2-17

Представители PN -типов неповторных формул в базисе $\{\&, \vee, \bar{}, \oplus\}$

h	№ п/п	Представитель типа формул	h	№ п/п	Представитель типа формул
1	1	1	4	10	$(1+1)1+1$
	2	2		11	$(1\oplus 1)1+1$
	2	$1+1$		12	$2+1+1$
	3	$1\oplus 1$		13	$1+1+1+1$

3	1	3		14	$1\oplus 1+1+1$
	2	$(1\oplus 1)1$		15	$2\oplus 1+1$
	3	$(1\oplus 1)1$			$(1+1)\oplus 1+1$
	4	$2+1$		16	$1\oplus 1\oplus 1+1$
	5	$1+1+1$		17	$3\oplus 1$
	6	$1\oplus 1+1$			$(1+1+1)\oplus 1$
	7	$2\oplus 1$		18	$(1+1)1\oplus 1$
	8	$(1+1)\oplus 1$			$(2+1)\oplus 1$
4	1	4		19	$(1\oplus 1)1\oplus 1$
	2	$(1+1)2$		20	$(1\oplus 1+1)+1$
	3	$(1\oplus 1)2$			$(2\oplus 1)\oplus 1$
	4	$(2+1)1$		21	$(1+1)\oplus 1\oplus 1$
	5	$(1+1+1)1$		22	$1\oplus 1\oplus 1\oplus 1$
	6	$(1\oplus 1+1)1$		23	$3\oplus 2$
	7	$(2+1)1$			$(1+1)(1+1)$
		$[(1+1)\oplus 1]1$		24	$2\oplus 2$
	8	$(1\oplus 1\oplus 1)1$			$(1+1)\oplus (1+1)$
	9	$3+1$		25	$2\oplus (1+1)$
				26	$(1\oplus 1)(1\oplus 1)$
		27	$1\oplus 1+1\oplus 1$		
		28	$2+1\oplus 1$		
		29	$(1+1)(1\oplus 1)$		

Например, формулы $f_1 = x_1x_2 \oplus x_3$ и $f_2 = (x_1 \vee x_2) \oplus x_3$ принадлежат одному PN -типу, так как $x_1x_2 \oplus x_3 = (\overline{x_1 \vee x_2}) \oplus \overline{x_3}$. К одному типу принадлежат также формулы $f_3 = (x_1 \oplus x_2)x_3 \oplus x_4$ и $f_4 = (x_1 \oplus x_2 \vee x_3) \oplus x_4$, так как $(x_1 \oplus x_2)x_3 \oplus x_4 = (\overline{x_1 \oplus x_2 \vee x_3}) \oplus \overline{x_4}$.

При этом справедливо утверждение: если формулы F_i и F_i' принадлежат одному PN -типу, то и формулы $F_1 \oplus F_2$ и $\overline{F_1} \oplus \overline{F_2}$ также принадлежат: одному PN -типов.

В работе [72] для $h = 1 \div 6$ найдено число представителей PN -типов неповторных формул в базисе $\{\&, \vee, \overline{}, \oplus\}$, а авторами получено соотношение для определения значений $B_3(h)$ - верхней оценки числа PW -типов неповторных формул в этом базисе для любых h . Значения $B_2(h)$ и $B_3(h)$ для $h \leq 10$ приведены в табл. 2-16.

В табл. 2-17 приведены результаты табулирования представителей PN -типов неповторных формул в рассматриваемом базисе при $h = 1 \div 4$, что необходимо для построения модулей, универсальных в этом классе формул.

Таким образом, ознакомившись с этой главой, можно сделать вывод, что неплохо иметь набор модулей, реализующих все формулы определенной длины. Еще лучше иметь один настраиваемый модуль, реализующий все формулы определенной длины. Однако изложенный материал дает возможность оценить и объем «бедствия», которое ждет, если погнаться за полной универсальностью и очень большими значениями K .

Проведенное табулирование показывает, что число неповторных формул растет, хотя и неизмеримо медленнее, чем число формул вообще, но все-таки достаточно быстро. Поэтому, вероятно, разумно ограничиться значениями $K = 6 \div 8$, а также, по возможности, остаться в рамках PN -классификации.

Ограничив K , мы приходим к задаче: как строить схемы, соответствующие формулам, число букв в которых превышает K ? Прежде чем привести ее решение (это будет сделано в гл. 4), займемся разработкой метода построения настраиваемых логических модулей.

Настраиваемые модули из функциональных элементов

3-1. ПОРОЖДАЮЩИЕ ФУНКЦИИ. СВОЙСТВА И ХАРАКТЕРИСТИКИ

Выше мы уже говорили, что основная цель создания НЛМ - сокращение номенклатуры составляющих схем при повышении уровня интеграции. Платой за достигаемый эффект при этом является избыточность по числу наружных выводов и внутренних элементов.

С теоретической точки зрения построить НЛМ - это значит так объединить между собой заданные функции, чтобы затем путем настройки можно было получить любую из этих функций.

Например, широко используют такие функции от двух переменных, как «И» и «ИЛИ», т. е. x_1x_2 , $x_1 \vee x_2$. Как их объединить? Это зависит от выбранного способа настройки (классификация способов приведена в предыдущей главе).

Простейший вариант - механическое объединение функций. Он малоинтересен, так как при этом у соответствующего модуля отсутствует функциональная интеграция: сложность модуля по числу внешних выводов и числу составляющих совпадает с этими характеристиками у объединяемых элементов.

Другой вариант - настройка путем выбора нужного выхода, в случае если входные выводы соответствующих элементов объединены.

Третий вариант - настройка по входам. Выходной вывод один, а входных три: один настроечный и два информационных. При этом модуль выполняет функцию $y = z_1z_2 \vee z_3$.

Легко заметить, что при подаче сигналов $z_1 = x_1, z_2 = x_2, z_3 = 0$ получим первую из требуемых функций $y = x_1x_2$, а при подаче $z_1 = x, z_2 = 1, z_3 = x_3$ - вторую функцию $y = x_1 \vee x_2$. Аналогичным будет результат в случае приравнивания входов: $z_1 = z_2 = x_1, z_3 = x_2$. Может сложиться впечатление, что настройка по входам ничего не дает по сравнению с настройкой по выходам: в обоих случаях число внешних выводов равно четырем, а внутренняя сложность модуля определяется суммой объединяемых элементов «И» и «ИЛИ». Однако так получилось лишь в частном случае, из-за того, что объединяемые функции не имеют общих составных частей. Если добавить к ним еще одну функцию, например $\overline{x_1x_2} \vee \overline{x_1x_2}$, то настройка выбором выхода потребует при наилучшем построении схемы пять внешних выводов и четыре двухвходовых элемента. Настройка же по входам также потребует четырех внутренних элементов, но число внешних выводов будет меньше - четыре вместо пяти.

Функция, выполняемая модулем, в этом случае имеет вид:

$$y = \overline{z_1z_2z_3} \vee z_1z_2z_3.$$

$$\text{При } z_3 = 1; \quad z_1 = \overline{x_1}; \quad z_2 = \overline{x_2} \quad y = x_1 \vee x_2;$$

$$\text{При } z_3 = 0; \quad z_1 = x_1; \quad z_2 = x_2 \quad y = x_1x_2;$$

$$\text{При } z_3 = x_2; \quad z_1 = z_2 = x_1 \quad y = \overline{x_1x_2} \vee \overline{x_1x_2};$$

При настройке по входам применяются следующие операции: подача на определенные входы констант 0 и 1, приравнивание входов друг к другу, приравнивание к инверсии и подача на информационные входы нужного сочетания прямых и инверсных значений входных переменных.

Если последнее возможно, то должна использоваться *PN*-классификация; если допустима только перестановка входных переменных без их инвертирования, то – *P*-классификация, и, наконец, при жесткой связи с источниками информации приходится объединять заданные функции, даже если отличие между ними состоит лишь в порядке следования букв.

С возрастанием числа объединяемых функций относительные преимущества настройки по входам возрастают, даже если общие части отсутствуют.

Пусть, например, в списке имеется сто функций от десяти переменных. Тогда при настройке по выходам потребуется сто десять внешних выводов, а при самой элементарной настройке по входам (без использования операции приравнивания входов и при полном разделении информационных и настроечных входов): $10 + \lceil \log_2 100 \rceil = 17$ внешних выводов, т. е. имеет место выигрыш в 6,5 раза!

Иногда варианты настройки по входам и выходам объединяют. Объясняется это тем, что от одних и тех же переменных могут потребоваться одновременно разные функции. Настройка по

выходам это обеспечит, в то время как настройка по входам - нет. Чаще всего требуется одновременно использовать прямое и

Таблица 3-1

**Оценки числа входов модулей,
универсальных в классе всех булевых функций
от n переменных и имеющих два выхода - прямой и инверсный**

n	2	3	4	5	6	7	8	9	10
Нижняя оценка	3	5	6	10	17	32	62	119	230
Верхняя оценка	3	5	7	16	28	39	72	137	266

инверсное значения функции и поэтому у модулей в этих случаях предусматривают два выхода: y и \bar{y} .

Функция, которую выполняет объединенный модуль при использовании настройки по входам, носит название порождающей (ПФ). Она порождает, в первую очередь, требуемый список функций, а кроме того, еще ряд функций, которые также могут оказаться полезными. При этом порожденные функции зависят от меньшего числа переменных, чем ПФ.

Различные ПФ обладают различной порождающей способностью. В литературе [16, 34, 37] были исследованы признаки, по которым можно найти наиболее эффективные функции, порождающие максимальное число подфункций. Однако, взяв одну из таких функций за основу при построении модуля, мы можем ошибиться, так как подфункций она порождает много, но, возможно, не тех, которые нам требуются. Проверка и выбор «правильной» функции связаны с большим перебором, затруднительным даже при использовании ЦВМ.

Подход, состоящий в выборе нужной ПФ из числа наиболее эффективных, применялся при построении модулей, универсальных в классе всех булевых функций от двух, трех и четырех аргументов [9, 69]. При большем числе переменных удалось установить только верхние и нижние оценки числа внешних выводов универсальных модулей, приведенные в табл. 3-1.

Для модулей этого типа была найдена также асимптотическая оценка числа входов, которая имеет следующий вид [25, 52]:

$$M(n) = \frac{2^n}{\log_2 n}. \quad (3-1)$$

Как было отмечено выше, подход, основанный на исчерпывающем исследовании логических возможностей булевых функций, связан с большим перебором, поэтому в литературе рассматривались также процедуры построения модулей без использования перебора или с меньшим его объемом, приводящие обычно к построению модулей с большим числом внешних выводов.

Наиболее простой метод построения модулей основан на том факте, что произвольная булева функция представима в виде

$$f(x_1, \dots, x_n) = \sum_{i=0}^{n-1} x_1^{i_1} \dots x_n^{i_n} f(i_1, \dots, i_n), \quad (3-2)$$

где

$$x_j^{i_j} = \begin{cases} x_j & \text{при } i_j = 1; \\ \bar{x}_j & \text{при } i_j = 0; \end{cases}$$

На основе этого соотношения может быть построен модуль, универсальный в классе всех булевых функций от n переменных, имеющий число внешних выводов

$$M(n) = 2^n + n + 1. \quad (3-3)$$

Этот модуль имеет 2^n настроечных входов, на которые должны подаваться только константы 0 и 1.

Если допустить подачу на настроечные входы не только констант, но и переменных, то число внешних выводов универсальных модулей может быть уменьшено [74]. Это достигается при построении модуля на основе соотношения

$$f(x_1, \dots, x_n) = \sum_{j=0}^{2^{n-1}-1} x_1^{i_1} \dots x_{n-1}^{i_{n-1}} f_i(i_1, \dots, i_{n-1}, x_n). \quad (3-4)$$

Число внешних выводов такого модуля

$$M(n) = 2^{n-1} + (n-1) + 1 = 2^{n-1} + n. \quad (3-5)$$

В этой же работе впервые для класса модулей из ФЭ был предложен метод построения модулей, универсальных в классе всех булевых функций из двух составляющих.

В изложенных подходах основное внимание уделялось построению модулей, универсальных в классе всех булевых функций. Рассмотрим методы построения модулей, реализующих путем настройки заданный набор функций.

Модуль, реализующий путем настройки N заданных булевых функций n переменных, может быть построен на основе соотношения

$$F(x_1, \dots, x_{n+d}) = \sum_{i=1}^N x_{n+d}^{j_{n+d}} \dots x_{n+1}^{j_{n+1}} f_i(x_n, \dots, x_1), \quad (3-6)$$

где $d = \lceil \log_2 N \rceil$ - символ округления до ближайшего большего целого числа.

Информационные и настроечные входы этого модуля независимы, а его настройка осуществляется путем подачи констант. Число внешних выводов такого модуля

$$M = \lceil \log_2 N \rceil + n - 1. \quad (3-7)$$

Для уменьшения числа внешних выводов необходимо отказаться, хотя бы частично, от жесткого разделения информационных и настроечных входов. В предельном случае любой вход может являться как информационным, так и настроечным:

$$F(x_1, \dots, x_{n+d}) = \sum x_{n+d}^{j_{n+d}} \dots x_1^{j_1} c_i(j_{n+d}, \dots, j_1), \quad (3-8)$$

где

$$c_i(j_{n+d}, \dots, j_1) = \begin{cases} 0; \\ 1. \end{cases}$$

Из этого соотношения следует, что построение такого модуля связано с нахождением значений c_i и d .

Э. А. Якубайтисом [59] был предложен метод построения модулей с минимальным числом выводов, реализующих путем настройки заданный набор из N булевых функций, базирующийся на соотношении (3-8). Сущность метода состоит в отборе на каждом шаге конъюнкций определенного ранга, включаемых в порождающую функцию, и в нахождении настроек, обеспечивающих выделение из получаемой порождающей функции заданных функций n переменных. Метод позволяет строить модули с минимальным числом внешних выводов, вследствие того что поиск оптимального значения выполняется «снизу вверх» от конъюнкций ранга $n+1$. Однако использование этого метода ограничивается относительно небольшими значениями N и n , так как в противном случае объем перебора становится чрезвычайно большим.

Наряду с общими подходами к построению модулей, изложенными выше, в настоящее время из литературы известны работы, посвященные модулям, универсальным в некоторых классах булевых функций.

В работах [18, 74] рассматривались модули, универсальные в классе симметрических функций n переменных. При этом было установлено, что число внешних выводов таких модулей равно $2n$, а их сложность линейно зависит от числа переменных.

В работе [60] были рассмотрены модули, универсальные в классе пороговых функций. Построение этих модулей выполнялось с использованием ЦВМ. При этом было установлено, что существует модуль с

восьмью входами, универсальный в указанном классе функций от пяти переменных, и модуль с десятью входами, универсальный в том же классе функций от шести переменных.

В работах [12, 44] были предложены подходы, на основе которых была разработана структура многофункционального модуля, реализующего путем настройки 26. неповторных в базе {&, ∨, ¬} формул из восьми букв. Другой тип многофункциональных модулей, реализующих путем настройки некоторые неповторные формулы в базе {&, ∨, ¬} из четырех букв, выпускается в настоящее время в виде микросхемы, входящей в серию 501 [55].

К сожалению, большинство методов, изложенных в указанных выше работах, носят частный характер, не имеют достаточного математического обеспечения и неприменимы поэтому в общем случае. Перейдем к изложению метода, позволяющего устранить эти недостатки.

3-2. ПОСТРОЕНИЕ МОДУЛЕЙ С МИНИМАЛЬНЫМ ЧИСЛОМ ВНЕШНИХ ВЫВОДОВ

Ограничимся рассмотрением модулей, настраиваемых по входам с помощью следующих операций: а) фиксация входов константами ($x_i = 0; 1$); б) отождествление входов ($x_j = x_i$); в) антиотождествление входов ($x_j = \bar{x}_i$).

Входы, с помощью которых осуществляется настройка, образуют настроечное поле, остальные - информационное. Иногда настроечное поле жестко отделяют от информационного: например, из пяти входов первые три используют всегда для подачи логических переменных, а два остальных - только для настройки.

Иногда разделение на настроечные и информационные входы носит условный характер - в каждом конкретном случае тот или иной входной вывод может служить для настройки.

Второй из указанных способов «богаче», так как появляется возможность различных вариантов выбора информационных и настроечных входов.

Пример. Пусть задана формула

$$y = \bar{x}_1(x_2 \vee \bar{x}_3) \vee x_1 \bar{x}_2 x_3$$

Если в этой формуле настроечной переменной считать всегда только x_1 , то получим две подфункции от двух аргументов: $y = x_2 \vee \bar{x}_3$ при $x_1 = 0$ и $y = \bar{x}_2 x_3$ при $x_1 = 1$. Если же считать, что настроечными могут быть как x_1 , так и x_2 , то получим три подфункции: $y = x_2 \vee \bar{x}_3$ при $x_1 = 0$, $y = \bar{x}_2 x_3$ при $x_1 = 1$ и $y = \bar{x}_1 x_3 \vee x_1 x_3$ при $x_2 = 0$.

Как правило, жесткое разделение информационного и настроечного полей выступает как ограничение, связанное с требованиями технологии или выбором структуры. В каких случаях оно имеет место, мы рассмотрим в дальнейшем. А сейчас попробуем глубже разобраться в процессе порождения подфункций. Что именно происходит с ПФ, когда мы фиксируем или отождествляем (антиотождествляем) входы?

Предположим, что ПФ найдена и для нее построена таблица истинности. Условимся в таблице всегда располагать настроечные переменные на позициях старших разрядов. Пусть число информационных переменных равно a , а настроечных d . Тогда при

Таблица 3-2

Число настроек $\bar{N}(d;b)$

d	b									
	0	1	2	3	4	5	6	7	8	9
1	2	1								
2	4	6	1							
3	8	28	12	1						
4	16	120	100	20	1					
5	32	496	720	260	30	1				
6	64	2016	4816	2800	560	42	1			
7	128	8128	30912	27216	8400	1064	56	1		
8	256	32640	193600	248640	111216	21168	1848	72	1	

9	512	130816	1194240	2182720	1360800	365232	44640	3000	90	1
---	-----	--------	---------	---------	---------	--------	-------	------	----	---

фиксации всех настроечных переменных константами от столбца значений ПФ останется подфрагмент длиной 2^a . При этом число различных подфрагментов равно 2^d .

Каждый подфрагмент - это не что иное, как одна из остаточных функций в разложении Шеннона [57], которое производится по настроечным аргументам.

Пример. Пусть задана функция

$$y = x_1 x_2 \vee x_3$$

Предположим, что x_1 - настроечная переменная; тогда разложение Шеннона по переменной x_1 имеет следующий вид:

$$y = \bar{x}_1 x_3 \vee x_1 (x_2 \vee x_3).$$

При этом имеют место два подфрагмента: первый x_3 , а второй $x_2 \vee x_3$. Как раз такие подфункции и получаются при соответствующей настройке: $y = x_3$ при $x_1 = 0$ и $y = x_2 \vee x_3$ при $x_1 = 1$.

Теперь рассмотрим случай, когда фиксируется только часть переменных настроечного поля, а остальные b переменных остаются свободными. При такой операции осуществляется выбор и объединение некоторых подфрагментов. Они также образуют «остаточную функцию». Тем самым мы можем не только выделять подфрагменты, но и объединять их между собой. Каждый вариант выделения и объединения - это новая настройка, порождающая функцию, но не обязательно новую, так как среди подфрагментов могут быть и одинаковые.

Таблица 3-3

Число настроек $N(d;b)$

d	b								
	0	1	2	3	4	5	6	7	8
1	2	1							
2	4	5	1						
3	8	19	9	1					
4	16	65	55	14	1				
5	32	211	285	125	20	1			
6	64	665	1551	910	245	27	1		
7	128	2059	6069	5901	2380	434	35	1	
8	256	6305	26335	35574	20181	5118	714	44	1

Оценки числа вариантов настройки. Число возможных вариантов настройки определяется числом переменных в настроечном поле d , числом свободных переменных b и числом используемых операций настройки.

В работе [52] найдена формула для подсчета числа настроек $\bar{N}(d;b)$ порождающей функции, содержащей d переменных, для реализации остаточной функции b переменных при использовании операций фиксации, отождествления и антиотождествления:

$$\bar{N}(d;b) = 2^{d-b} S(d+1;b+1), \quad (3-9)$$

где $S(d+1;b+1)$ - число Стирлинга второго рода [49].

В табл. 3-2 приведены значения $\bar{N}(d;b)$, подсчитанные по формуле (3-9) при $d \leq 9, b \leq 9$.

В той же работе найдена также формула для подсчета числа настроек $N(d;b)$ ПФ, содержащей d переменных, для реализации остаточной функции b переменных при использовании операций фиксации и отождествления переменных:

$$N(d;b) = S(d+2;b+2) - S(d+1;b+2). \quad (3-10)$$

В табл. 3-3 приведены значения $N(d;b)$, подсчитанные по формуле (3-10) при $d \leq 8, b \leq 8$.

Приведенные соотношения и количественные характеристики позволяют определить нижние оценки числа переменных g в ПФ, обеспечивающие число настроек, большее или равное числу заданных

функций N , зависящих от b переменных. Эти оценки могут быть получены в случае, если в ПФ информационные и настроечные переменные совпадают.

При этом $g = d$ и справедливы неравенства

$$\bar{N}(d;b) = 2^{d-b} S(d+1; b+1) \geq N(b), \quad (3-11)$$

$$N(d;b) = S(d+2; b+2) - S(d+1; b+2) \geq N(b). \quad (3-12)$$

Объединение подфрагментов для получения остаточных функций. Если из d переменных настроечного поля b являются свободными, то с их помощью могут объединяться 2^b подфрагментов длины 2^a . Номенклатура объединяемых подфрагментов зависит от вида операции настройки. Обозначим операцию объединения столбцов значений подфрагментов знаком $*$.

Пример. Пусть задана булева функция от $g = 2$ переменных. Предположим, что $a = 0, d = 2, b = 1$. При этом длина каждого подфрагмента

Таблица 3-4

Объединение подфрагментов при $a = 0, d = 2, b = 1$

x_2	x_1	F	$x_2 = 0$	$x_2 = 1$	$x_1 = 0$	$x_1 = 1$	$x_2 = x_1$	$x_2 = \bar{x}_1$
0	0	A	A	-	A	-	A	-
0	1	B	B	-	-	B	-	B
1	0	C	-	C	C	-	-	C
1	1	D	-	D	-	D	D	-

равна $2^a = 1$, а их число $2^d = 4$. Обозначим их буквами A, B, C, D . Выполняя операции настройки, получим шесть различных фрагментов, состоящих из $2^b = 2$ подфрагментов (табл. 3-4):

$$A * B; \quad C * D; \quad A * C; \quad B * D; \quad A * D; \quad B * C.$$

Предположим, что $a = 1, d = 1, b = 0$. При этом длина подфрагментов $2^a = 2$, а их число $2^d = 2$. Обозначим подфрагменты буквами G, K . Выполняя операции настройки, получим два фрагмента, совпадающих с соответствующими подфрагментами ($2^b = 1$) (табл. 3-5).

Определим условия, при выполнении которых имеется возможность объединить любые 2^b подфрагментов ПФ из общего их числа 2^b в остаточную функцию таким образом, чтобы подфрагмент, имеющий меньший номер в столбце значений ПФ, был размещен в остаточной функции над подфрагментом с большим номером. Для того чтобы иметь возможность выполнить указанное объединение подфрагментов при использовании операций фиксации переменных константами и отождествлении переменных, необходимо выполнение соотношения

$$N(d;b) \geq C_{2^d}^{2^b}, \quad (3-13)$$

Таблица 3-5

Объединение подфрагментов при $a = 1, d = 1, b = 0$

x_2	x_1	F	$x_2 = 0$	$x_2 = 1$
0	0	G	G	-
0	1			
1	0	K	-	K
1	1			

а при использовании также операции «Антиотождествление» требуется, чтобы выполнялось соотношение

$$\bar{N}(d;b) \geq C_{2^d}^{2^b}. \quad (3-14)$$

Исследуем, при каких значениях b эти соотношения выполняются.

Случай 1: $b = 0$. Использование операции фиксации переменных константами позволяет выбрать любой одиночный подфрагмент столбца значений ПФ. Действительно, из табл. 3-3 следует, что число настроек при $b = 0$ равно 2^d .

Случай 2: $b = 1$. Использование двух операций — фиксации переменных константами и отождествления переменных — не позволяет объединить любую пару подфрагментов ПФ в остаточную функцию так, чтобы подфрагмент с меньшим номером всегда размещался над подфрагментом с большим номером.

Пусть задано настроечное поле ПФ из d переменных. Объединение любых двух из 2^d подфрагментов ПФ так, как это было указано выше, возможно $C_{2^d}^2$ способами. С другой стороны, число различных вариантов настройки, объединяющих подфрагменты, которое обеспечивает это настроечное поле для одной свободной переменной к при использовании указанных операций, на основании (3-10) равно

$$N(d;1) = S(d+2;3) - S(d+1;3). \quad (3-15)$$

При этом справедливость неравенства

$$C_{2^d}^2 < S(d+2;3) - S(d+1;3) \quad (3-16)$$

доказывает высказанное утверждение.

Пример. Пусть $d = 3$, тогда $C_{2^3}^2 = 28$. При этом, так как $N(3, 2) = 19$,

то не удастся объединить любую пару подфрагментов ПФ в остаточную. Докажем следующее утверждение. Использование трех операций настройки, указанных в начале параграфа, позволяет объединить любую пару подфрагментов ПФ в остаточную так, что подфрагмент с меньшим номером всегда будет размещен над подфрагментом с большим номером.

Пусть задано настроечное поле ПФ из d переменных. Объединение любых двух из 2^d подфрагментов ПФ так, как это указано в формулировке утверждения, возможно $C_{2^d}^2$ способами. С другой стороны, число различных вариантов настройки для одной свободной переменной и использования указанных в формулировке утверждения операций на основании (3-10) равно

$$\overline{N}(d;1) = 2^{d-1} S(d+1;2). \quad (3-17)$$

При этом справедливость равенства

$$C_{2^d}^2 = 2^{d-1} S(d+1;2) \quad (3-18)$$

доказывает утверждение.

Введенные выше операции настройки позволяют объединить пару подфрагментов в указанном порядке. Для обеспечения возможности их объединения в произвольном порядке введем четвертую операцию настройки - замену переменных их инверсиями. При этом справедливо следующее утверждение: использование четырех операций настройки позволяет объединить любую пару подфрагментов ПФ в остаточную функцию при произвольном порядке их расположения в ней.

Случай 3: $b = 2$. Пусть в настроечном поле имеются две свободные переменные и используются три операции настройки; тогда для $d = 3$ число настроек $\overline{N}(3;2) = 12$. Следовательно, в этом случае осуществляется выбор и объединение четырех подфрагментов из восьми в двенадцати различных сочетаниях. С другой стороны, общее число сочетаний из 2^d подфрагментов по четыре при условии, что в столбце значений остаточной функции каждый подфрагмент ПФ с меньшим номером будет расположен над подфрагментом ПФ с большим номером, равно $C_{2^d}^4$. При $d = 3$ $C_8^4 = 70$. Поэтому уже при $d = 3$ не удастся выбрать и объединить в остаточную функцию произвольные четыре подфрагмента ПФ.

В общем случае объединение произвольных четырех подфрагментов ПФ при использовании трех операций настройки не удастся, так как

$$C_{2^d}^4 > \overline{N}(d;2) \quad (3-19)$$

Например, при $d = 6$ $C_{64}^4 = 635376$, в то время как $N(6;2) = 4816$.

Так как объединение произвольных четырех подфрагментов невозможно, а выбор определенных сочетаний из четырех подфрагментов связан с большим перебором, то в дальнейшем будем рассматривать лишь случаи, когда объединению подлежат два подфрагмента.

Построение порождающей функции настраиваемого модуля.

При построении модулей из ФЭ основным критерием является число внешних выводов, которое при заданных функциональных возможностях требуется минимизировать [32].

Рассмотрим следующую задачу. Пусть дан список из N булевых функций, каждая из которых существенно зависит от n переменных: $f_1(x_n, \dots, x_1), f_2(x_n, \dots, x_1), \dots, f_N(x_n, \dots, x_1)$. Требуется построить булеву функцию $F(z_g, \dots, z_1)$, зависящую от возможно меньшего числа переменных $g > n$, которая способна в результате выполнения над ней операций настройки порождать любую из N упомянутых функций.

Под возможно меньшим числом переменных g будем подразумевать не абсолютный минимум, а минимум, достижимый при реальном объеме перебора.

Из изложенного ранее следует, что для построения ПФ надо решить две основные задачи: а) сформировать на основании списка f_1, f_2, \dots, f_N определенный состав подфрагментов; б) разместить подфрагменты в рациональном порядке, позволяющем выбирать и объединять их путем настройки. Решение этих задач позволяет математически обеспечить метод построения ПФ.

Простейший вариант решения - использовать в качестве подфрагментов сами функции f_1, f_2, \dots, f_N . При этом порядок их расположения может быть произвольным, а настроечное поле отделено от информационного. Число настроечных переменных в этом случае определяется из соотношения

$$d = \lceil \log_2 N \rceil. \quad (3-20)$$

ПФ при таком подходе имеет следующий вид:

$$F = \bigvee_{i=1}^N z_{n+d}^{\sigma_d} z_{n+d-1}^{\sigma_{d-1}} \dots z_{n+1}^{\sigma_1} f_i(z_n, \dots, z_1), \quad (3-21)$$

где

$$z_{n+j}^{\sigma_j} = \begin{cases} z_j & \text{при } \sigma_j = 1; \\ \bar{z}_j & \text{при } \sigma_j = 0. \end{cases}$$

Пример. Построить ПФ для следующего набора функций:

$$\begin{aligned} f_1 &= x_1 x_2 x_3; \\ f_2 &= x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 x_2 x_3; \\ f_3 &= x_1 \vee x_2 \vee x_3; \\ f_4 &= x_1 x_2 \vee x_1 x_3 \vee x_2 x_3. \end{aligned}$$

Поскольку $N = 4$, то $d = \log_2 4 = 2$.

Введем настроечные переменные x_4 и x_5 ; тогда функция

$$F_1 = \bar{x}_5 \bar{x}_4 f_1 \vee \bar{x}_5 x_4 f_2 \vee x_5 \bar{x}_4 f_3 \vee x_5 x_4 f_4$$

является порождающей.

Точно так же порождающими являются функции с любым другим расположением подфрагментов, например функция

$$F_2 = \bar{x}_5 \bar{x}_4 f_2 \vee \bar{x}_5 x_4 f_4 \vee x_5 \bar{x}_4 f_1 \vee x_5 x_4 f_3$$

Несмотря на то, что модули, соответствующие этим функциям, имеют одинаковое число внешних выводов, их элементная сложность различна: некоторые формулы минимизируются в большей, некоторые в меньшей степени.

В изложенном способе формирования ПФ вид порождаемых функций f_i никак не влияет на число входов. Оно всегда составляет $g' = n + \lceil \log_2 N \rceil$. В то же время в функциях f_i могут быть общие составные части.

Возникает предположение, что наличие одинаковых составных частей в различных функциях может быть использовано с целью сокращения суммарного числа подфрагментов. В самом деле, какой смысл

дважды вносить в ПФ идентичные «половины» двух функций? Может быть, достаточно записать в ПФ каждую из составных частей функции f_i всего один раз, а потом, объединяя их, составлять требуемые функции?

Предположение это оправданно. Только, к сожалению, в большинстве случаев не удастся в качестве «строительного материала» ПФ использовать составные части, меньшие чем «половина» функции f_i , так как выше мы показали, что любые два подфрагмента можно путем настройки объединить в любом порядке. Этот вывод по отношению к четырем, восьми и большему числу подфрагментов несправедлив.

Следовательно, для формирования ПФ достаточно: а) определить номенклатуру подфрагментов длиной: 2^{n-1} , из которых могут быть сформированы заданные функции; б) объединить подфрагменты в одну функцию, которая и будет соответствовать искомой ПФ.

Формирование подфрагментов осуществляется путем разложения каждой из функций f_i по одной из переменных. Так как допустимо произвольное переименование входных переменных, то будем считать, что разложение производится по переменной x_n :

$$f_i(x_n, \dots, x_1) = \bar{x}_n f_i(0, x_{n-1}, \dots, x_1) \vee x_n f_i(1, x_{n-1}, \dots, x_1) = \bar{x}_n \varphi_j'(x_{n-1}, \dots, x_1) \vee x_n \varphi_j''(x_{n-1}, \dots, x_1). \quad (3-22)$$

При табличной форме задания, когда переменные имеют порядок расположения x_n, x_{n-1}, \dots, x_1 и разложение функции выполняется по переменной x_n , одному подфрагменту соответствует верхняя половина столбца значений функции, а другому - нижняя половина.

Разложим каждую из N функций по одной переменной и предположим, что суммарное число различных подфрагментов у всех этих N функций равно r . *Два подфрагмента будем называть различными, если порядки расположения нулей и единиц в их столбцах значений не совпадают.*

Необходимым условием для того, чтобы некоторая функция была порождающей, является входимость каждого из r подфрагментов, указанных выше, в столбец значений этой функции по крайней мере один раз. Выполнение этого условия обеспечивает минимальную номенклатуру подфрагментов, которую должен иметь столбец значений функции, претендующей на роль порождающей для заданного набора из N функций.

Если подфрагменты в функции могут быть выбраны в любом сочетании и порядке, то их минимальная номенклатура является и достаточной, так как ни одна из N заданных функций не должна иметь двух одинаковых подфрагментов длиной 2^{n-1} ввиду того, что в противном случае она зависела бы от одной из переменных несущественно.

Вторым вопросом, решаемым при построении ПФ, является определение порядка расположения найденных подфрагментов требуемой номенклатуры в столбце значений ПФ.

Пусть имеется столбец значений некоторой функции, разделенный на $2^{\lceil \log_2 r \rceil}$ позиций длиной 2^{n-1} , в котором значения функций еще не проставлены. Разместим на некоторых позициях столбца в произвольном порядке r различных подфрагментов длиной 2^{n-1} , а остальные $2^{\lceil \log_2 r \rceil} - r$ позиций заполним произвольно.

Из сказанного выше следует, что сформированная таким образом функция является порождающей для заданного набора N функций при использовании операций настройки четырех типов, так как любая пара подфрагментов, номенклатура которых совпадает с номенклатурой подфрагментов заданных функций, может быть выбрана в любом сочетании и порядке, образуя остаточные функции, совпадающие с заданными.

Таким образом, при указанном способе формирования ПФ число настроечных переменных $d = \lceil \log_2 r \rceil$, а число информационных переменных $a = n - 1$. При этом общее число переменных ПФ

$$g = \lceil \log_2 r \rceil + n - 1, \quad (3-23)$$

а число различных ПФ

$$v = (2^{\lceil \log_2 r \rceil}). \quad (3-24)$$

При аналитической форме представления построенная ПФ имеет вид

$$(3-25)$$

где

Эффективность метода. Для того чтобы ПФ, построенная при помощи изложенного метода, содержала меньшее число переменных по сравнению с их числом, получаемым при объединении в ПФ самих функций, необходимо выполнение неравенства

$$g'' < g'. \quad (3-26)$$

Подставляя в (3-26) значения g' и g'' , получим
(3-27)

Решение этого неравенства имеет следующий вид:

$$M_{\text{обл}} \leq \dots \quad (3-28)$$

Полученная оценка для числа подфрагментов может быть в ряде случаев улучшена. Для этого определим, какое число различных подфрагментов длиной 2^{n-1} содержится во всех булевых функциях n переменных. Значение этой величины

$$G_{\text{ш.х}} = 2^{2^n - 1}. \quad (3-29)$$

На основе объединения полученных соотношений можно утверждать, что число различных фрагментов z должно удовлетворять неравенству

$$z < \min\{2^{\lceil \log_2 N \rceil}; 2^{2^n - 1}\} \quad (3-30)$$

Полученное соотношение является условием, при выполнении которого удастся построить ПФ, зависящую от числа переменных, меньшего чем g' . В противном случае число переменных в ПФ равно g' .

Пример. При $n = 3, N = 40 \quad z < 16$; при $n = 3, N = 4 \quad z < 4$.

Минимизация числа переменных порождающей функции.

Выше было показано, что при применении предлагаемого метода число переменных в ПФ g'' при $n = \text{const}$ зависит только от числа различных подфрагментов z в N заданных функциях. Поэтому проблема минимизации числа переменных в ПФ сводится к сокращению числа различных подфрагментов. Можно утверждать, что число различных подфрагментов в функциях, зависящих от n переменных, в общем случае определяется двумя факторами: а) выбором множества N функций, для которых ищется ПФ; б) выбором переменных, по которым осуществляется разложение заданных функций.

Для заданных N функций число различных подфрагментов определяется только вторым фактором. Выбор переменных, обеспечивающих минимизацию числа подфрагментов, в общем случае связан с перебором.

Введем критерий, использование которого позволяет выбрать требуемые переменные без перебора. Этот критерий является эвристическим, однако его применение при построении модулей показало, что он достаточно эффективен. Будем использовать при выборе переменной, по которой следует проводить разложение, вес производной булевой функции по переменной [43].

Производной булевой функции, зависящей от n переменных, по переменной x_k называется функция $n - 1$ переменных, равная сумме по модулю 2 остаточных исходной функции при $x_k = 0$, и $x_k = 1$, т.е.

$$\frac{\delta f(x_1, \dots, x_n)}{\delta x_k} = f(x_k = 0) \oplus f(x_k = 1) \quad (3-31)$$

Весом производной $P\left(\frac{\delta f}{\delta x_k}\right)$ называют число наборов, на которых она принимает значение, равное единице.

Пусть некоторая функция f зависит от нескольких переменных, и в том числе от x_k , и x_i . Будем выполнять разложение по x_k раньше, чем по x_i , если справедливо неравенство

$$P\left(\frac{\delta f}{\delta x_k}\right) > P\left(\frac{\delta f}{\delta x_i}\right) \quad (3-32)$$

Введенный критерий предложен для разложения функций, принадлежащих классу произвольных функций. Покажем, что выбор этого критерия является естественным. Пусть задана функция, принадлежащая классу пороговых. Произведем разложение этой функции по переменной,

обладающей наибольшим весом [7]. Для этого класса функций выбор переменной, по которой должно выполняться разложение, по любому из двух указанных критериев приводит обычно к одному и тому же результату. Если же некоторая функция является симметричной по переменным x_k и x_i , то разложение может выполняться по любой из этих переменных.

Если задано несколько булевых функций, каждая из которых зависит от n переменных, то их первоначально необходимо переобозначить так, чтобы при разложении по какой-либо одной переменной общее число различных фрагментов было минимальным. Для достижения этого воспользуемся следующей процедурой: а) в каждой из заданных формул определяем веса производных по всем переменным; б) производим переобозначение переменных в каждой формуле так, что переменная с большим весом производной будет иметь и больший порядковый номер; в) раскладываем все заданные формулы по переменной с наибольшим порядковым номером.

Использование этой процедуры, как правило, позволяет уменьшить число различных подфрагментов в системе заданных функций по сравнению с разложением в соответствии с исходной нумерацией переменных, хотя это положение и не удается доказать.

Пример. Пусть заданы функции $f_1 = v_1 v_2 v_3$, $f_2 = v_1 (v_2 \vee v_3)$, $f_3 = v_1 v_3 \vee v_2$, $f_4 = v_1 \vee v_2 \vee v_3$. Определим нумерацию переменных, при котором число различных подфрагментов будет минимальным. Составим таблицу истинности для заданных функций (табл. 3-6).

Таблица 3-6

Таблица истинности функций $f_1 \div f_4$

v_3	v_2	v_1	f_1	f_2	f_3	f_4	v_3	v_2	v_1	f_1	f_2	f_3	f_4
0	0	0	0	0	0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	1	1	0	0	0	1	1
0	1	1	0	1	1	1	1	1	1	1	1	1	1

Таким образом, заданный набор функций содержит шесть различных подфрагментов: $\varphi_1 = 0000$; $\varphi_2 = 0001$; $\varphi_3 = 0011$; $\varphi_4 = 0101$; $\varphi_5 = 0111$, $\varphi_6 = 1111$. Попытаемся уменьшить значение r путем переобозначения переменных. Функции f_1 и f_4 симметричны по всем переменным; в f_2 наибольшим весом производной обладает переменная v_1 , а в f_3 — переменная v_2 . Поэтому переобозначим переменные следующим образом: $f_1 = x_1 x_2 x_3$, $f_2 = (x_1 \vee x_2) x_3$, $f_3 = x_1 x_2 \vee x_3$, $f_4 = x_1 \vee x_2 \vee x_3$.

Из таблицы истинности этих функций (табл. 3-7) следует, что число различных подфрагментов после переобозначения при разложении по переменной x_3 равно четырем: $\varphi_1 = 0000$;

$$\varphi_2 = 0001; \varphi_3 = 0111; \varphi_4 = 1111.$$

Таблица 3-7

Минимизация числа подфрагментов

x_3	x_2	x_1	f_1	f_2	f_3	f_4	x_3	x_2	x_1	f_1	f_2	f_3	f_4
0	0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	1	0	0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	0	1	1	1	0	0	1	1	1
0	1	1	0	0	1	1	1	1	1	1	1	1	1

Таким образом, использование предложенной процедуры позволило уменьшить число различных подфрагментов и тем самым сократить число входов модуля.

Минимизация элементной сложности модуля. Эта задача сводится к задаче упрощения ПФ, зависящей от фиксированного числа переменных, и, в частности, к минимизации числа букв в ней. Так как ПФ при большом числе подфрагментов в объединяемых функциях зависит от сравнительно большого числа переменных, то нахождение формулы, минимальной по числу букв, для представления ПФ в аналитическом виде с помощью классического аппарата минимизации является чрезвычайно громоздким и трудоемким. При некоторых условиях, которые будут определены ниже, удастся упростить процесс минимизации. Запишем ПФ модуля в следующем виде:

$$F_2(z_g, \dots, z_1) = \bigvee_{i=1}^v z_g^{\sigma_g} \dots z_{n+1}^{\sigma_{n+1}} \psi_i(z_n, \dots, z_1), \quad (3-33)$$

где

$$v = 2^{\lceil \log_2 r \rceil - 1}, g = \lfloor \log_2 r \rfloor + n - 1.$$

Такое представление ПФ может быть получено в результате объединения соседних пар подфрагментов. При этом

$$\psi_i(z_n, \dots, z_1) = z_n \varphi'(z_{n-1}, \dots, z_1) \vee \bar{z}_n \varphi''(z_{n-1}, \dots, z_1). \quad (3-34)$$

Поскольку подфрагменты представляют собой функции и поэтому могут быть проминимизированы отдельно, направим усилия на минимизацию числа букв в той части конъюнкции в выражении (3-33), которая состоит из настроечных переменных. Для этого ПФ будем искать в форме, отличающейся от формы выражения (3-33) тем, что буквы с инверсиями, соответствующие настроечным переменным, в ней исключены. При этом указанные части конъюнкций соответствуют всевозможным сочетаниям из $\lfloor \log_2 r \rfloor - 1$ переменных настроечного поля без инверсий, взятым по $0, 1, 2, \dots, \lfloor \log_2 r \rfloor - 1$ переменным, а общее число конъюнкций в этой форме не изменяется по сравнению с их числом в выражении (3-33) и равно $2^{\lfloor \log_2 r \rfloor - 1}$. Назовем эту форму ПФ минимизированной. Для того чтобы такая форма представления ПФ была эквивалентна F_2 , фрагменты ψ_i должны удовлетворять системе соотношений, получаемой из условий равенства ПФ, представленных в этих формах. Сравнение этих выражений будем выполнять в табличной форме.

Для формализованного обозначения соотношений между фрагментами ψ_i и ψ_j введем в рассмотрение функцию импликации. В первой главе мы уже пользовались функцией, задаваемой таблицей истинности вида табл. 3-8, которая называется импликацией x_1 в x_2 .

Говорят, что функция f_i имплицирует f_j ($f_i \rightarrow f_j$), если столбец значений таблицы истинности f_j «покрывает» соответствующий столбец f_i . При этом, если значение f_i на некотором наборе равно нулю, то и значение f_j на том же наборе также равно нулю; если значение f_j на некотором наборе равно единице, то значение f_i на

Таблица истинности функции импликации Таблица 3-8

x_1	x_2	f
0	0	1
0	1	1
1	0	0
1	1	1

этом наборе может быть произвольным. Поэтому, в частности, можно утверждать, что $v_1 v_2 \rightarrow v_1 \vee v_2$. Отметим, что если при построении ПФ без учета ее элементной сложности подфрагменты могли размещаться в столбце значений произвольно, то при нахождении ПФ в минимизированной форме, системе импликаций которой должны удовлетворять фрагменты, накладываются ограничения на порядок их размещения, несмотря на возможность использования четырех разновидностей операций настройки, рассмотренных выше.

Пример. Пусть число подфрагментов, зависящих от трех переменных, в ПФ равно восьми. Тогда эта функция имеет вид

$$F_1(z_6, \dots, z_1) = \bar{z}_6 \bar{z}_5 \bar{z}_4 \varphi_1 \vee \bar{z}_6 \bar{z}_5 z_4 \varphi_2 \vee \bar{z}_6 z_5 \bar{z}_4 \varphi_3 \vee \bar{z}_6 z_5 z_4 \varphi_4 \vee z_6 \bar{z}_5 \bar{z}_4 \varphi_5 \vee z_6 \bar{z}_5 z_4 \varphi_6 \vee z_6 z_5 \bar{z}_4 \varphi_7 \vee z_6 z_5 z_4 \varphi_8.$$

Приведем это выражение к виду (3-33):

$$F_2(z_6, \dots, z_1) = \bar{z}_6 \bar{z}_5 (\bar{z}_4 \varphi_1 \vee z_4 \varphi_2) \vee \bar{z}_6 z_5 (\bar{z}_4 \varphi_3 \vee z_4 \varphi_4) \vee z_6 \bar{z}_5 (\bar{z}_4 \varphi_5 \vee z_4 \varphi_6) \vee z_6 z_5 (\bar{z}_4 \varphi_7 \vee z_4 \varphi_8) = \bar{z}_6 \bar{z}_5 \psi_1 \vee \bar{z}_6 z_5 \psi_2 \vee z_6 \bar{z}_5 \psi_3 \vee z_6 z_5 \psi_4. \quad (3-36)$$

Определим условия для ψ_i , при выполнении которых F_2 может быть представлена в минимизированной форме:

$$F_3(z_6, \dots, z_1) = \psi_1 \vee z_5 \psi_2 \vee z_6 \psi_3 \vee z_5 z_6 \psi_4. \quad (3-37)$$

Составим таблицы истинности для функции F_2 и F_3 и определим условия, выполнение которых необходимо для эквивалентности этих функций (табл. 3-9)

Таблица 3-9

Условия эквивалентности функции F_2 и F_3

z_6	z_5	F_2	F_3
0	0	ψ_1	ψ_1
0	1	ψ_2	$\psi_1 \vee \psi_2$
1	0	ψ_3	$\psi_1 \vee \psi_3$
1	1	ψ_4	$\psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4$

При этом $F_3 = F_2$, если выполняется система соотношений вида:

$$\psi_1 = \psi_1; (3-38)$$

$$\psi_1 \vee \psi_2 = \psi_2; (3-39)$$

$$\psi_1 \vee \psi_3 = \psi_3; (3-40)$$

$$\psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4 = \psi_4; (3-41)$$

Из соотношения (3-41) следует, что фрагмент ψ_1 должен быть построен в результате объединения подфрагментов ψ_i и ψ_j , так чтобы выполнялась система импликации

$$\psi_1 \rightarrow \psi_4; \psi_2 \rightarrow \psi_4; \psi_3 \rightarrow \psi_4, (3-42)$$

Т. е. столбец значений ψ_4 должен «покрывать» столбцы значений ψ_1, ψ_2, ψ_3 .

Из соотношений (3-39) и (3-40) следует, что фрагмент ψ_1 должен удовлетворять системе импликаций

$$\psi_1 \rightarrow \psi_2; \psi_1 \rightarrow \psi_3, (3-43)$$

т. е. столбцы значений ψ_2 и ψ_3 должны «покрывать» столбец значений ψ_1 . Таким образом, фрагмент ψ_1 должен быть выбран так, чтобы он «покрывался» всеми остальными фрагментами, а фрагмент ψ_4 сам должен «покрывать» все остальные фрагменты. Фрагменты ψ_2 и ψ_3 строятся так, чтобы соответствующие им функции были наиболее простыми. При этом, как следует из табл. 3-9, порядок расположения фрагментов должен быть следующим: $\psi_1, \psi_2, \psi_3, \psi_4$.

Следовательно, если фрагменты могут быть сформированы так, что соотношения (3-42) и (3-43) выполняются, то ПФ может быть представлена в минимизированном виде (3-37). После этого ПФ может быть проминимизирована еще больше с помощью классических методов. Если хотя бы одна из импликаций не выполняется, то ПФ записывается в виде (3-36), после чего она также минимизируется классическими методами.

Процедура построения порождающей функции. На основании результатов, изложенных выше, может быть предложена процедура построения ПФ модуля, реализующего путем настройки заданный набор из N булевых функций, каждая из которых существенно зависит от n переменных.

1. В каждой из N заданных функций определяем веса производных по всем переменным.
2. Производим переобозначение переменных в каждой функции так, что переменная с большим весом производной будет иметь и больший порядковый номер.
3. Раскладываем все функции по переменной с наибольшим порядковым номером.
4. Формируем список различных подфрагментов φ_i , содержащихся в N функциях набора.
5. Подсчитываем число различных подфрагментов φ_i . При этом число переменных в ПФ определяется из соотношения

$$g = \lceil \log_2 r \rceil + n - 1. (3-44)$$

6. Если полученные подфрагменты φ , могут быть объединены таким образом, что выполняется система соотношений вида

$$\left. \begin{aligned} \psi_1 &= \psi_1; \\ \psi_2 &= \psi_1 \vee \psi_2; \\ \psi_3 &= \psi_1 \vee \psi_3; \\ &\dots\dots\dots \\ \psi_t &= \psi_1 \vee \psi_2 \vee \dots \vee \psi_t, \end{aligned} \right\} \quad (3-45)$$

где

$$\psi_i(z_n, \dots, z_1) = z_n \varphi'(z_{n-1}, \dots, z_1) \vee z_n \varphi''(z_{n-1}, \dots, z_1),$$

то искомая ПФ может быть представлена в виде

$$F(z_g, \dots, z_1) = \varphi_1 \vee z_{n+1} \psi_2 \vee z_{n+2} \psi_3 \vee \dots \vee z_g \psi_j \vee z_{n+1} z_{n+2} \psi_k \vee \dots \vee z_{n+1} z_{n+2} \dots z_g \psi_t. \quad (3-46)$$

Если найденные подфрагменты φ_i , не могут быть объединены так, чтобы выполнялось каждое из соотношений приведенной системы, то ПФ представляется в виде

$$F(z_g, \dots, z_1) = \bigvee_{i=1}^t z_g^{\sigma_g} \dots z_n^{\sigma_n} \varphi_i(z_{n-1}, \dots, z_1) = \bigvee_{i=1}^t z_g^{\sigma_g} \dots z_{n+1}^{\sigma_{n+1}} \varphi_j(z_n, \dots, z_1), \quad (3-47)$$

где

$$v = 2^{\lceil \log_2 r \rceil}; t = 2^{\lceil \log_2 r \rceil - 1}.$$

7. Осуществляем минимизацию найденной функции с помощью классических методов с целью уменьшения элементной сложности настраиваемого модуля.
8. Составляем таблицу истинности из переменных настроечного поля z_g, \dots, z_n , в столбце значений которой проставляем виденные подфрагменты φ_i , в порядке их расположения в ПФ.
9. Каждую из N заданных функций представляем путем объединения столбцов значений подфрагментов в виде

$$f_i = \varphi_j \times \varphi_k. \quad (3-48)$$

10. Таблицу настроек искомого модуля для реализаций функций, однотипных с заданными, определяем на основе сопоставления каждого из выражений вида $f_i = \varphi_j \times \varphi_k$ и таблицы истинности настроечного поля, с помощью которой осуществляется выбор и объединение подфрагментов φ_j и φ_k .

3-3. ПРИМЕР ПОСТРОЕНИЯ ПОРОЖДАЮЩЕЙ ФУНКЦИИ НАСТРАИВАЕМОГО МОДУЛЯ

Воспользуемся предложенной процедурой для нахождения структуры модуля, универсального в классе формул в базисе $\{\&, \vee, \bar{}\}$ из пяти букв.

Предполагая, что прямые и инверсные входы *III* равнодоступны и возможна перестановка входных переменных, для построения ПФ модуля воспользуемся *PN*-классификацией. При этом в ПФ достаточно объединить следующие 24 представителя *PN-типов* неповторных формул в базисе $\{\&, \vee, \bar{}\}$ из пяти букв:

1. $f_1 = x_1 x_2 x_3 x_4 x_5$;
2. $f_2 = (x_1 \vee x_2) x_3 x_4 x_5$;
3. $f_3 = (x_1 x_2 \vee x_3) x_4 x_5$;
4. $f_4 = (x_1 \vee x_2 \vee x_3) x_4 x_5$;
5. $f_5 = (x_1 x_2 x_3 \vee x_4) x_5$;
6. $f_6 = [(x_1 \vee x_2) x_3 \vee x_4] x_5$;
7. $f_7 = (x_1 x_2 \vee x_3 \vee x_4) x_5$;
8. $f_8 = (x_1 \vee x_2 \vee x_3 \vee x_4) x_5$;
9. $f_9 = x_1 x_2 x_3 x_4 \vee x_5$;
10. $f_{10} = (x_1 \vee x_2) x_3 x_4 \vee x_5$;
11. $f_{11} = (x_1 x_2 \vee x_3) x_4 \vee x_5$;
12. $f_{12} = (x_1 \vee x_2 \vee x_3) x_4 \vee x_5$;
13. $f_{13} = x_1 x_2 x_3 \vee x_4 \vee x_5$;
14. $f_{14} = (x_1 \vee x_2) x_3 \vee x_4 \vee x_5$;
15. $f_{15} = x_1 x_2 \vee x_3 \vee x_4 \vee x_5$;
16. $f_{16} = x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5$;
17. $f_{17} = x_1 x_2 \vee x_3 x_4 \vee x_5$;
18. $f_{18} = (x_1 \vee x_2)(x_3 \vee x_4) \vee x_5$;
19. $f_{19} = (x_1 x_2 \vee x_3 x_4) x_5$;
20. $f_{20} = (x_1 \vee x_2)(x_3 \vee x_4) x_5$;
21. $f_{21} = x_1 x_2 x_3 \vee x_4 x_5$;
22. $f_{22} = (x_1 \vee x_2) x_3 \vee x_4 x_5$;
23. $f_{23} = (x_1 x_2 \vee x_3)(x_4 x_5)$;
24. $f_{24} = (x_1 \vee x_2 \vee x_3)(x_4 \vee x_5)$.

Нумерация переменных в этих формулах выбрана так, что переменные с большим весом булевой производной имеют больший порядковый номер. Выбранное правило нумерации обеспечивает сокращение числа различных подфрагментов для заданного набора формул и в конечном итоге приводит к уменьшению числа внешних выводов в модуле.

Для нахождения различных подфрагментов составим таблицу истинности этих формул (табл. 3-10). Заданный набор формул содержит 16 различных подфрагментов длиной 16 (табл. 3-11).

$$\begin{aligned}
 F(z_8, \dots, z_1) = & \overline{z_8 z_7 z_6 z_5 \varphi_\alpha} \vee \overline{z_8 z_7 z_6 z_5 \varphi_\beta} \vee \overline{z_8 z_7 z_6 z_5 \varphi_j} \vee \overline{z_8 z_7 z_6 z_5 \varphi_u} \vee \\
 & \overline{z_8 z_7 z_6 z_5 \varphi_\omega} \vee \overline{z_8 z_7 z_6 z_5 \varphi_v} \vee \overline{z_8 z_7 z_6 z_5 \varphi_p} \vee \overline{z_8 z_7 z_6 z_5 \varphi_e} \vee \overline{z_8 z_7 z_6 z_5 \varphi_t} \vee \\
 & \overline{z_8 z_7 z_6 z_5 \varphi_h} \vee \overline{z_8 z_7 z_6 z_5 \varphi_k} \vee \overline{z_8 z_7 z_6 z_5 \varphi_n} \vee \overline{z_8 z_7 z_6 z_5 \varphi_r} \vee \overline{z_8 z_7 z_6 z_5 \varphi_\lambda} \vee \\
 & \overline{z_8 z_7 z_6 z_5 \varphi_h} \vee \overline{z_8 z_7 z_6 z_5 \varphi_k} = \overline{z_8 z_7 z_6 (z_5 \varphi_\alpha \vee z_5 \varphi_\beta)} \vee \overline{z_8 z_7 z_6 (z_5 \varphi_j \vee z_5 \varphi_u)} \vee \\
 & \overline{z_8 z_7 z_6 (z_5 \varphi_\omega \vee z_5 \varphi_v)} \vee \overline{z_8 z_7 z_6 (z_5 \varphi_p \vee z_5 \varphi_e)} \vee \overline{z_8 z_7 z_6 (z_5 \varphi_t \vee z_5 \varphi_s)} \vee \overline{z_8 z_7 z_6 (z_5 \varphi_k \vee z_5 \varphi_n)} \vee \overline{z_8 z_7 z_6 (z_5 \varphi_r \vee z_5 \varphi_\lambda)} \vee \\
 & \overline{z_8 z_7 z_6 (z_5 \varphi_h \vee z_5 \varphi_k)}. (3-49)
 \end{aligned}$$

Таким образом, минимизированной формой ПФ является формула вида

$$\begin{aligned}
 F(z_8, \dots, z_1) = & \overline{z_8 z_7 z_6 \varphi_1} \vee \overline{z_8 z_7 z_6 \varphi_2} \vee \overline{z_8 z_7 z_6 \varphi_4} \vee \overline{z_8 z_7 z_6 \varphi_5} \vee \\
 & \overline{z_8 z_7 z_6 \varphi_6} \vee \overline{z_8 z_7 z_6 \varphi_7} \vee \overline{z_8 z_7 z_6 \varphi_8}.
 \end{aligned} \tag{3-50}$$

Определим соотношение для ψ_1 , при выполнении которых F может быть представлена в виде

$$F'(z_8, \dots, z_1) = \psi_1 \vee z_6 \psi_2 \vee z_7 \psi_3 \vee z_7 z_6 \psi_4 \vee z_8 \psi_5 \vee z_8 z_6 \psi_6 \vee z_8 z_7 \psi_7 \vee z_8 z_7 z_6 \psi_8. \tag{3-51}$$

Составим таблицы истинности для функции F и F' и определим условия, при которых эти функции эквивалентны (табл. 3-12).

Таблица 3-11

Таблица истинности подфрагментов

x	x_3	x_2	x_1	φ_1	φ_2	φ_3	φ_4	φ_5	φ_6	φ_7	φ_8	φ_9	φ_{10}	φ_{11}	φ_{12}	φ_{13}	φ_{14}	φ_{15}	φ_{16}
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	1	1	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1
0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	1
0	1	0	1	0	0	0	0	0	1	1	1	0	1	0	1	0	1	1	1
0	1	1	0	0	0	0	0	0	1	1	1	0	1	0	1	0	1	1	1
0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1
1	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	1	1
1	0	0	1	0	0	0	1	0	0	1	1	0	0	0	1	1	1	1	1
1	0	1	0	0	0	0	1	0	0	1	1	0	0	0	1	1	1	1	1
1	0	1	1	0	0	0	1	0	0	1	1	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1
1	1	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Таблица 3-12

Условия минимизации ПФ

z_8	z_7	z_6	F	F'
0	0	0	ψ_1	ψ_1
0	0	1	ψ_2	$\psi_1 \vee \psi_2$
0	1	0	ψ_3	$\psi_1 \vee \psi_3$
0	1	1	ψ_4	$\psi_1 \vee \psi_3$
1	0	0	ψ_4	$\psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4$
1	0	1	ψ_5	$\psi_1 \vee \psi_5$
1	1	0	ψ_5	$\psi_1 \vee \psi_5$
1	1	1	ψ_6	$\psi_1 \vee \psi_2 \vee \psi_5 \vee \psi_6$
			ψ_7	$\psi_1 \vee \psi_3 \vee \psi_5 \vee \psi_7$
			ψ_8	$\psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4 \vee \psi_5 \vee \psi_6 \vee \psi_7 \vee \psi_8$

При этом $F' = F$, если имеет место система соотношений

$$\psi_1 = \psi_1; \quad (3-52) \quad \psi_1 \vee \psi_5 = \psi_5; \quad (3-56)$$

$$\psi_1 \vee \psi_2 = \psi_2; \quad (3-53) \quad \psi_1 \vee \psi_2 \vee \psi_5 \vee \psi_6 = \psi_6; \quad (3-57)$$

$$\psi_1 \vee \psi_3 = \psi_3; \quad (3-54) \quad \psi_1 \vee \psi_3 \vee \psi_5 \vee \psi_7 = \psi_7; \quad (3-58)$$

$$\psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4 = \psi_4; \quad (3-55) \quad \psi_1 \vee \psi_2 \vee \psi_3 \vee \psi_4 \vee \psi_5 \vee \psi_6 \vee \psi_7 \vee \psi_8 = \psi_8; \quad (3-59)$$

Из этих соотношений следует, что

$$\psi \rightarrow \psi_2; \psi_1 \rightarrow \psi_3; \psi_1 \rightarrow \psi_4; \psi_1 \rightarrow \psi_5; \psi_1 \rightarrow \psi_6; \psi_1 \rightarrow \psi_7; \psi_1 \rightarrow \psi_8, \quad (3-60)$$

т.е. столбец значений ψ_1 должен «показываться» любым из оставшихся семи фрагментов ψ_i .

Пусть

$$\psi_1 = \varphi_1 \times \varphi_2 = z_1 z_2 z_3 z_4 z_5, \quad (3-61)$$

Соотношения (3-53)-(3-55) порождают импликацию

$$\psi_2 \vee \psi_3 \rightarrow \psi_4, \quad (3-62)$$

Соотношения (3-53),(3-56),(3-57)-импликацию

$$\psi_2 \vee \psi_5 \rightarrow \psi_6, \quad (3-63)$$

а соотношения (3-54), (3-56), (3-58)-импликацию

$$\psi_3 \vee \psi_5 \rightarrow \psi_7, \quad (3-64)$$

Из соотношения (3-59) следует, что

$$\psi_1 \rightarrow \psi_8; \psi_2 \rightarrow \psi_8; \psi_3 \rightarrow \psi_8; \psi_4 \rightarrow \psi_8; \psi_5 \rightarrow \psi_8; \psi_6 \rightarrow \psi_8; \psi_7 \rightarrow \psi_8; \quad (3-65)$$

т.е. столбец значений ψ_8 должен «покрывать» любой из оставшихся семи фрагментов ψ_i .

Пусть

$$\psi_8 = \varphi_{15} \times \varphi_{16} = z_1 \vee z_2 \vee z_3 \vee z_4 \vee z_5, \quad (3-66)$$

Если выбрать (табл. 3-13)

$$\left. \begin{aligned} \psi_2 &= \varphi_3 \times \varphi_4 = z_1 z_2 z_3 \vee z_4 z_5; \\ \psi_3 &= \varphi_5 \times \varphi_6 = (z_1 \vee z_2) z_3 (z_4 \vee z_5); \\ \psi_4 &= \varphi_7 \times \varphi_8 = (z_1 \vee z_2)(z_3 \vee z_4) \vee z_4 z_5; \end{aligned} \right\} (3-67)$$

то будет выполняться импликация (3-62).

При выборе

$$\left. \begin{aligned} \psi_5 &= \varphi_9 \times \varphi_{10} = (z_1 z_2 \vee z_3)(z_4 \vee z_5); \\ \psi_6 &= \varphi_{11} \times \varphi_{12} = z_1 z_2 \vee z_3 z_4 \vee (z_3 \vee z_4) z_5; \end{aligned} \right\} (3-68)$$

удовлетворяется импликация (3-63).

И, наконец, выбрав

$$\psi_7 = \varphi_{13} \times \varphi_{14} = (z_1 \vee z_2 \vee z_3)(z_4 \vee z_5), \quad (3-69)$$

закончим формирование фрагментов (табл. 3-13).

При этом ПФ имеет вид

$$\begin{aligned} F(z_8, \dots, z_1) &= z_1 z_2 z_3 z_4 z_5 \vee (z_1 z_2 z_3 \vee z_4 z_5) z_6 \vee (z_1 \vee z_2) z_3 (z_4 \vee z_5) z_7 \vee \\ &\vee (z_1 z_2 \vee z_3)(z_4 \vee z_5) z_8 \vee (z_1 \vee z_2)(z_3 \vee z_4) z_6 z_7 \vee z_1 z_2 z_6 z_8 \vee \\ &\vee (z_1 \vee z_2)(z_3 \vee z_4) z_7 z_8 \vee (z_1 \vee z_2 \vee z_3 \vee z_4 \vee z_5) z_6 z_7 z_8. \end{aligned} \quad (3-71)$$

Таблица 3-13

Таблица объединения фрагментов

z_5	z_4	z_3	z_2	z_1	ψ_1	ψ_2	ψ_3	ψ_4	ψ_5	ψ_6	ψ_7	ψ_8
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	0	0	0	1	0	1
0	0	1	0	0	0	0	0	0	0	0	0	1
0	0	1	0	1	0	0	0	1	0	0	0	1
0	0	1	1	0	0	0	0	1	0	0	0	1
0	0	1	1	1	0	1	0	1	0	1	0	1
0	1	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	1	0	0	0	1	0	0	1	1
0	1	0	1	0	0	0	0	1	0	0	1	1
0	1	0	1	1	0	0	0	1	1	1	1	1
0	1	1	0	0	0	0	0	0	1	1	1	1
0	1	1	0	1	0	0	1	1	1	1	1	1
0	1	1	1	0	0	0	1	1	1	1	1	1
0	1	1	1	1	0	1	1	1	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	0	0	0	0	1	1
1	0	0	1	0	0	0	0	0	0	0	1	1
1	0	0	1	1	0	0	0	0	1	1	1	1
1	0	1	0	0	0	0	0	0	1	1	1	1
1	0	1	0	1	0	0	1	1	1	1	1	1
1	0	1	1	0	0	0	1	1	1	1	1	1
1	0	1	1	1	0	1	1	1	1	1	1	1
1	1	0	0	0	0	1	0	0	0	1	0	1
1	1	0	0	1	0	1	0	0	0	1	1	1
1	1	0	1	0	0	1	0	0	0	1	1	1
1	1	0	1	1	0	1	0	0	1	1	1	1
1	1	1	0	0	0	1	0	0	1	1	0	1
1	1	1	0	1	0	1	1	1	1	1	1	1
1	1	1	1	0	0	1	1	1	1	1	1	1
1	1	1	1	1	0	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1

Таблица 3-14

Таблица истинности переменных настроечного поля

z_8	z_7	z_6	z_5	F
0	0	0	0	φ_1
0	0	0	1	φ_2
0	0	1	0	φ_3
0	0	1	1	φ_4
0	1	0	0	φ_5
0	1	0	1	φ_6
0	1	1	0	φ_7
0	1	1	1	φ_8
1	0	0	0	φ_9
1	0	0	1	φ_{10}
1	0	1	0	φ_{11}
1	0	1	1	φ_{12}
1	1	0	0	φ_{13}
1	1	0	1	φ_{14}
1	1	1	0	φ_{15}
1	1	1	1	φ_{16}

Так как полученная формула достаточно сложна (она содержит 47 букв), продолжим процесс ее минимизации:

$$F(z_8, \dots, z_1) = z_1 z_2 (z_3 z_4 z_5 \vee z_6 z_8) \vee \{z_1 z_2 z_3 \vee z_4 z_5 \vee [(z_1 \vee z_2)(z_3 \vee z_4 \vee z_8)(z_3 \vee z_4 \vee z_5)z_8]z_7\}z_6 \vee \vee [(z_1 \vee z_2)(z_3 \vee z_8)z_7 \vee (z_1 z_2 \vee z_3)z_8](z_4 \vee z_5) \cdot (3 - 72)$$

Эта формула содержит 34 буквы.

Составим таблицу истинности для переменных настроечного поля ПФ с целью определения настроек, порождающих заданные функции (табл. 3-14).

Таблица 3-15

Таблица настроек для K=5

Настройка	Объединяемые фрагменты	Реализуемая формула
$z_8 = 0, z_7 = 0, z_6 = 0$	$\varphi_1 \times \varphi_2$	$F_1 = z_1 z_2 z_3 z_4 z_5$
$z_8 = 0, z_6 = 0, z_5 = 0$	$\varphi_1 \times \varphi_5$	$F_2 = (z_1 \vee z_2)z_3 z_4 z_7$
$z_7 = 0, z_6 = 0, z_5 = 0$	$\varphi_1 \times \varphi_9$	$F_3 = (z_1 z_2 \vee z_3)z_4 z_8$
$z_8 = z_7, z_6 = 0, z_5 = 0$	$\varphi_1 \times \varphi_{13}$	$F_4 = (z_1 \vee z_2 \vee z_3)z_4 z_7$
$z_8 = 0, z_7 = 0, z_6 = z_5$	$\varphi_1 \times \varphi_4$	$F_5 = (z_1 z_2 z_3 \vee z_4)z_5$
$z_8 = 0, z_7 = z_6 = z_5$	$\varphi_1 \times \varphi_8$	$F_6 = [(z_1 \vee z_2)z_3 \vee z_4]z_5$
$z_8 = z_6 = z_5, z_7 = 0$	$\varphi_1 \times \varphi_{12}$	$F_7 = (z_1 z_2 \vee z_3 \vee z_4)z_5$
$z_8 = z_7 = z_6, z_5 = 0$	$\varphi_1 \times \varphi_{15}$	$F_8 = (z_1 \vee z_2 \vee z_3 \vee z_4)z_6$
$z_8 = z_7 = z_6, z_5 = 1$	$\varphi_2 \times \varphi_{16}$	$F_9 = z_1 z_2 z_3 z_4 \vee z_6$
$z_8 = z_6 = z_5, z_7 = 1$	$\varphi_5 \times \varphi_{16}$	$F_{10} = (z_1 \vee z_2)z_3 z_4 \vee z_5$
$z_8 = 1, z_7 = z_6 = z_5$	$\varphi_9 \times \varphi_{16}$	$F_{11} = (z_1 z_2 \vee z_3)z_4 \vee z_5$
$z_8 = 1, z_7 = 1, z_6 = z_5$	$\varphi_{13} \times \varphi_{16}$	$F_{12} = (z_1 \vee z_2 \vee z_3)z_4 \vee z_5$
$z_8 = z_7, z_6 = 1, z_5 = 1$	$\varphi_4 \times \varphi_{16}$	$F_{13} = z_1 z_2 z_3 \vee z_4 \vee z_7$
$z_7 = 1, z_6 = 1, z_5 = 1$	$\varphi_8 \times \varphi_{16}$	$F_{14} = (z_1 \vee z_2)z_3 \vee z_4 \vee z_8$
$z_8 = 1, z_6 = 1, z_5 = 1$	$\varphi_{12} \times \varphi_{16}$	$F_{15} = z_1 z_2 \vee z_3 \vee z_4 \vee z_7$
$z_8 = 1, z_7 = 1, z_6 = 1$	$\varphi_{15} \times \varphi_{16}$	$F_{16} = z_1 \vee z_2 \vee z_3 \vee z_4 \vee z_5$
$z_8 = 1, z_7 = z_6, z_6 = 1$	$\varphi_{11} \times \varphi_{16}$	$F_{17} = z_1 z_2 \vee z_3 z_4 \vee z_5$
$z_8 = z_5, z_7 = 1, z_6 = 1$	$\varphi_7 \times \varphi_{16}$	$F_{18} = (z_1 \vee z_2)(z_3 \vee z_4) \vee z_5$
$z_8 = z_6, z_7 = 0, z_5 = 0$	$\varphi_1 \times \varphi_{11}$	$F_{19} = (z_1 z_2 \vee z_3 z_4)z_6$
$z_8 = 0, z_7 = z_6, z_5 = 0$	$\varphi_1 \times \varphi_7$	$F_{20} = (z_1 \vee z_2)(z_3 \vee z_4)z_6$
$z_8 = 0, z_7 = z_6, z_5 = 0$	$\varphi_3 \times \varphi_4$	$F_{21} = z_1 z_2 z_3 \vee z_4 z_5$
$z_8 = 0, z_7 = 0, z_6 = 1$	$\varphi_6 \times \varphi_8$	$F_{22} = (z_1 \vee z_2)z_3 \vee z_4 z_6$
$z_8 = 0, z_7 = 1, z_5 = 1$	$\varphi_9 \times \varphi_{10}$	$F_{23} = (z_1 z_2 \vee z_3)(z_4 \vee z_5)$
$z_8 = 1, z_7 = 0, z_6 = 0$	$\varphi_{13} \times \varphi_{14}$	$F_{24} = (z_1 \vee z_2 \vee z_3)(z_4 \vee z_5)$

Заданные формулы из пяти букв могут быть образованы в результате объединения следующих подфрагментов:

$$\begin{array}{llll}
 f_1 = \varphi_1 \times \varphi_2; & f_7 = \varphi_1 \times \varphi_{12}; & f_{13} = \varphi_4 \times \varphi_{16}; & f_{19} = \varphi_1 \times \varphi_{11}; \\
 f_2 = \varphi_1 \times \varphi_5; & f_8 = \varphi_1 \times \varphi_{15}; & f_{14} = \varphi_8 \times \varphi_{16}; & f_{20} = \varphi_1 \times \varphi_7; \\
 f_3 = \varphi_1 \times \varphi_9; & f_9 = \varphi_2 \times \varphi_{16}; & f_{15} = \varphi_{12} \times \varphi_{16}; & f_{21} = \varphi_3 \times \varphi_4; \\
 f_4 = \varphi_1 \times \varphi_{13}; & f_{10} = \varphi_5 \times \varphi_{16}; & f_{16} = \varphi_{15} \times \varphi_{16}; & f_{22} = \varphi_6 \times \varphi_8; \\
 f_5 = \varphi_1 \times \varphi_4; & f_{11} = \varphi_9 \times \varphi_{16}; & f_{17} = \varphi_{11} \times \varphi_{16}; & f_{23} = \varphi_9 \times \varphi_{10}; \\
 f_6 = \varphi_1 \times \varphi_8; & f_{12} = \varphi_{13} \times \varphi_{16}; & f_{18} = \varphi_7 \times \varphi_{16}; & f_{24} = \varphi_{13} \times \varphi_{14};
 \end{array}$$

Поэтому таблица настроек (табл. 3-15) искомого модуля для реализаций функций, однотипных с заданными, определяется на основе сопоставления каждого выражения вида $f_i = \varphi_j \times \varphi_k$ и таблицы истинности настроечного поля, с помощью которой осуществляется выбор и объединение подфрагментов φ_j и φ_k .

Таблица 3-16

Таблица настроек модуля для K=4

Настройка	Реализуемая формула
$z_8 = 0, z_7 = 0, z_6 = 0, z_5 = 1$	$F_1 = z_1 z_2 z_3 z_4$
$z_8 = 0, z_7 = 1, z_6 = 0, z_5 = 0$	$F_2 = (z_1 \vee z_2) z_3 z_4$
$z_8 = 1, z_7 = 0, z_6 = 0, z_5 = 0$	$F_3 = (z_1 z_2 \vee z_3) z_4$
$z_8 = 1, z_7 = 0, z_6 = 0, z_5 = 0$	$F_4 = (z_1 \vee z_2 \vee z_3) z_4$
$z_8 = 0, z_7 = 0, z_6 = 1, z_5 = 1$	$F_5 = z_1 z_2 z_3 \vee z_4$
$z_8 = 0, z_7 = 1, z_6 = 1, z_5 = 1$	$F_6 = (z_1 \vee z_2) z_3 \vee z_4$
$z_8 = 1, z_7 = 0, z_6 = 1, z_5 = 1$	$F_7 = z_1 z_2 \vee z_3 \vee z_4$
$z_8 = 1, z_7 = 1, z_6 = 1, z_5 = 0$	$F_8 = z_1 \vee z_2 \vee z_3 \vee z_4$
$z_8 = 1, z_7 = 0, z_6 = 1, z_5 = 0$	$F_9 = z_1 z_2 \vee z_3 z_4$
$z_8 = 0, z_7 = 1, z_6 = 1, z_5 = 0$	$F_{10} = (z_1 \vee z_2)(z_3 \vee z_4)$

Модули, разрабатываемые с помощью изложенного в § 3-2 метода, обладают характерным свойством, состоящим в том, что если модуль универсален в классе формул в некотором базисе из K букв, то он также универсален в том же классе формул из $K-1$ букв при условии, что его информационные и настроечные входы независимы.

Таким образом, модуль универсален при выполнении указанного условия для $K=4$ (табл. 3-16).

Порождающая функция модуля, универсального в классе формул в базисе $\{\&, \vee, \bar{}\}$ из трех букв, найденная с помощью той же процедуры, имеет вид

$$F(z_4, z_3, z_2, z_1) = (z_1 \vee z_2 \vee z_3) z_4 \vee z_1 z_2 z_3.$$

Приведем таблицу настроек этого модуля (табл. 3-17).

Настройка	Реализуемая формула
$z_4 = 0$	$F_1 = z_1 z_2 z_3$
$z_4 = 1$	$F_2 = z_1 \vee z_2 \vee z_3$
$z_3 = 0$	$F_3 = (z_1 \vee z_2) z_4$
$z_3 = 1$	$F_4 = z_1 z_2 \vee z_4$

3-4. Техническая реализация настраиваемых логических модулей

В состав некоторых серий интегральных микросхем, выпускаемых отечественной промышленностью в настоящее время, входят НЛМ, принципы построения которых близки принципам, излагаемым в настоящей работе. Рассмотрим функциональные и электрические характеристики этих модулей.

В состав серии 108 [51] входит НЛМ среднего уровня интеграции, реализованный на МОП-транзисторах в виде микросхемы, оформленной в прямоугольном стеклянном корпусе 401.14-1. Эта микросхема имеет следующие электрические характеристики:

Напряжение источника питания, В..... $-27 \pm 10\%$
 Потребляемая мощность, мВт, не более..... 100
 Потребляемый ток, мА, не более..... 2,6
 Амплитуда импульсов входного напряжения, В, не менее..... $-9,0$
 Напряжение логической единицы (выходное), В, не менее..... $-9,5$
 Напряжение логического нуля (выходное), В, не более..... $-0,7$
 Частота следования импульсов входного напряжения, кГц, не более.... 100

Бесповторные формулы из восьми букв, реализуемые модулем К1ЖЛ081 путем настройки

Способ настройки	F_1	F_2
$x_9 = 0, x_{10} = 0$	$F_1^1 = x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8$	$F_1^1 = x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8$
$x_9 = 0, x_{10} = 1$	$F_1^2 = x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7 x_8$	$F_1^2 = x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7 x_8$
$x_9 = 1, x_{10} = 0$	$F_1^3 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 \vee x_8$	$F_1^3 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 \vee x_8$
$x_9 = 1, x_{10} = 1$	$F_1^4 = x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_7 x_8$	$F_1^4 = x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_7 x_8$
$x_9 = 1, x_7 = 1$	$F_1^5 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_8 \vee x_{10}$	$F_1^5 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_8 \vee x_{10}$
$x_{10} = 1, x_7 = 1$	$F_1^6 = x_1 x_2 x_3 x_4 \vee x_5 x_6 x_8 \vee x_9$	$F_1^6 = x_1 x_2 x_3 x_4 \vee x_5 x_6 x_8 \vee x_9$
$x_9 = x_{10}, x_8 = 0$	$F_1^7 = x_1 x_2 x_3 x_4 x_5 x_6 x_7 \vee x_9$	$F_1^7 = x_1 x_2 x_3 x_4 x_5 x_6 x_7 \vee x_9$
$x_0 = 0, x_7 = x_8$	$F_1^8 = (x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7) x_{10}$	$F_1^8 = (x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7) x_{10}$
$x_9 = 0, x_4 = x_8$	$F_1^9 = (x_1 x_2 x_3 \vee x_5 x_6 x_7) x_4 x_{10}$	$F_1^9 = (x_1 x_2 x_3 \vee x_5 x_6 x_7) x_4 x_{10}$
$x_{10} = 0, x_7 = x_8$	$F_1^{10} = (x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7) x_9$	$F_1^{10} = (x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7) x_9$
$x_9 = x_{10}, x_7 = 0$	$F_1^{11} = (x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_8) x_9$	$F_1^{11} = (x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_8) x_9$
$x_9 = x_{10}, x_4 = x_8$	$F_1^{12} = (x_1 x_2 x_3 \vee x_5 x_6 \vee x_7 \vee x_8) x_9$	$F_1^{12} = (x_1 x_2 x_3 \vee x_5 x_6 \vee x_7 \vee x_8) x_9$
$x_{10} = 0, x_6 = x_8$	$F_1^{13} = [x_1 x_2 \vee x_3 x_4 \vee x_6 (x_5 \vee x_7)] x_9$	$F_1^{13} = [x_1 x_2 \vee x_3 x_4 \vee x_6 (x_5 \vee x_7)] x_9$

Модуль имеет десять логических входов, два выхода и реализует функции вида

$$F_1(x_1, \dots, x_{10}) = x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10} \vee (x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 x_8) x_9 x_{10} \vee (x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7 x_8) x_9 x_{10} \vee (x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_7 \vee x_8) x_9 x_{10}; (3-73)$$

$$F_2(x_1, \dots, x_{10}) = F_1(x_1, \dots, x_{10}).$$

Для определения функциональных возможностей модуля в классе формул, бесповторных в базисе $\{\&, \vee, \bar{}\}$, приравняем переменные друг другу и константам. Получившиеся при этом бесповторные формулы запишем в таблицу, формируя тем самым таблицу настроек модуля.

В табл. 3-18 приведены все бесповторные формулы максимальной длины (восемь букв), реализуемые модулем путем настроек, а в работе [6] указаны

Бесповторные формулы из четырех букв, реализуемые Модулем К1ЖЛ014 путем настройки

Управляющие входы			F	Арифметический полином
P	0	R		
0	0	0	$F_1 = A_1 A_2 \vee A_3 A_4$	$2 + 2$
0	0	1	$F_2 = A_1 A_2 A_3 A_4$	4
0	1	0	$F_3 = A_1 A_2 \vee A_3 \vee A_4$	$2 + 1 + 1$
0	1	1	$F_4 = A_1 A_2 (A_3 \vee A_4)$	$2(1 + 1)$
1	0	0	$F_5 = A_1 \vee A_2 \vee A_3 A_4$	$1 + 1 + 2$
1	0	1	$F_6 = (A_1 \vee A_2) A_3 A_4$	$(1 + 1)2$
1	1	0	$F_7 = A_1 \vee A_2 \vee A_3 \vee A_4$	$1 + 1 + 1 + 1$
1	1	1	$F_8 = (A_1 \vee A_2)(A_3 \vee A_4)$	$(1 + 1)(1 + 1)$

Все бесповторные формулы меньшей длины, получаемые из формул, приведенных в табл. 3-18. Анализ функциональных возможностей этого модуля, выполненный в § 5-1, позволил установить, что при использовании PN -классификации он универсален в классе произвольных формул в базисе $\{\&, \vee, \bar{}\}$ из пяти букв и в классе бесповторных ДНФ из шести букв.

Другой настраиваемый логический модуль входит в состав серии К501.

Микросхема К1ЖЛ014 состоит из трех одинаковых НЛМ с прямым и инверсным выходами, настраиваемыми общими входами P, θ, R [55].

Порождающая функция, описывающая поведение каждого из трех модулей на прямом выходе, имеет вид

$$F = \overline{[(A_1 \vee A_2)P \vee A_1 A_2 \vee [(A_3 \vee A_4)\theta \vee A_3 A_4]R \vee [(A_1 \vee A_2)P \vee A_1 A_2][(A_3 \vee A_4)\theta \vee A_3 A_4]}. (3-74)$$

Этот модуль позволяет реализовать путем настройки шесть представителей из десяти существующих типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из четырех букв (табл. 3-19).

Как следует из приведенных результатов, этот модуль не универсален в классе формул в базисе $\{\&, \vee, \bar{\quad}\}$ из четырех букв, так как не позволяет реализовать путем настройки представителей остальных четырех типов формул: $3+1; (1+1)1+1; (1+1+1)1; (2+1)1$. Однако этот модуль универсален в рассматриваемом классе формул из трех и менее букв.

Микросхема К1ЖЛ014 реализована на МОП-структурах и имеет следующие основные электрические параметры:

Выходные напряжения, В

логической единицы..... -9,5

логического нуля..... -1,0

Входное напряжение, В

логические единицы..... -8,5

логического нуля..... -2,0

Напряжение питания, В..... $-12,6 \pm 10\%$

Напряжение смещения, В... $-27 \pm 10\%$

Настраиваемые модули в настоящее время реализуются не только средствами микроэлектронной интегральной технологии. Так, в состав универсальной системы элементов промышленной пневмоавтоматики (УСЭППА) [11] входит одновыходное трехмембранное реле Р-3Ф, структура которого описывается функцией четырех переменных:

$$y = x_1(x_1 \vee \bar{x}_3) \vee \bar{x}_2 x_3 x_4. (3-75)$$

Исследование функциональных возможностей этого модуля показано, что он реализует путем настройки большое число формул, и в том числе следующие формулы из трех букв:

при $x_1 = 0$ $y_1 = \bar{x}_2 x_3 x_4 \equiv 3;$

при $x_1 = 1$ $y_2 = x_2 \vee \bar{x}_3 \vee x_4 \equiv 1+1+1;$

при $x_3 = 1$ $y_3 = x_1(x_2 \vee x_4) \equiv 1(1+1);$

при $x_4 = 0, x_4 = x_2$ $y'_3 = x_1(x_2 \vee \bar{x}_3) \equiv 1(1+1);$

при $x_2 = x_1$ $y_4 = x_1 \vee x_3 x_4 \equiv 1+2;$

при $x_4 = 1, x_4 = x_3$ $y'_4 = x_1 \vee \bar{x}_2 x_3 \equiv 1+2.$

Таким образом, можно утверждать, что рассматриваемый модуль универсален в классе формул в базисе $\{\&, \vee, \bar{\quad}\}$ из трех букв при использовании PN -классификации.

То, что мы изложили выше, отвечает на вопросы: «Какие модули делать?» и «Как их проектировать?» Но этого, конечно, недостаточно: необходимо еще знать, что делать с готовыми модулями, т.е. как построить из них схемы, обладающие минимальной элементной сложностью. Это и составляет предмет следующей главы.

Построение логических схем из настраиваемых логических модулей

4-1. Реализация схем в базисе настраиваемых модулей

Совершенствование элементарной базы систем логического управления в настоящее время невозможно без использования системного подхода. В основе такого подхода лежит одновременная разработка собственно НЛМ и их математического (логического) обеспечения, которое позволяет использовать широкие функциональные возможности таких модулей. При этом наибольшие преимущества достигаются при разработке серии НЛМ, совместимых не только по электрическим, технологическим и конструктивным параметрам, но и по логическому обеспечению.

Необходимо отметить, что до последнего времени разработка элементной базы шла независимо от создания методов ее использования, так как число различных типов элементов было незначительным и существующие теоретические методы синтеза были в основном достаточны для применения элементов, выпускаемых промышленностью. С появлением большого числа разнотипных модулей с широкими функциональными возможностями, к которым неприменимы классические методы синтеза, не удается достичь эффективного их использования без разработки соответствующего логического обеспечения.

В ходе разработки такого обеспечения должен быть предложен подход к использованию НЛМ, который позволит применять разнотипные модули и серии из них с помощью единой процедуры. Это должно обеспечить наряду со схемной унификацией и повышением уровня интеграции элементной базы унификацию в области ее применения.

В настоящей главе эти вопросы рассматриваются применительно к разработке логического обеспечения НЛМ, универсальных в классе формул. В основу излагаемого подхода положены следующие предпосылки.

Любой модуль, имеющий хотя бы два входа, можно рассматривать как настраиваемый, потому что, приравнивая часть входов нулю, единице или друг другу, будем получать настройки на различные остаточные подфункции.

Рациональное применение логических модулей требует предварительного исследования всех их функциональных возможностей в том классе функций, для которого соответствующий метод реализации обеспечит использование этих возможностей.

Мы надеемся, что нам уже удалось показать особую роль неповторных формул и соответствующих модулей. Напомним: модуль, реализующий неповторные формулы из K букв, выполняет также и повторные формулы из K букв, имеющие ту же структуру.

Таким образом, результаты, относящиеся к модулям, реализующим путем настройки неповторные формулы, обладают универсальностью. Поэтому при рассмотрении формул, подлежащих реализации, удобнее говорить просто о фрагментах (подформулах) длиной в K букв, а содержат ли эти фрагменты одинаковые аргументы или разные - для нас не имеет значения. Дальнейшее изложение ведется в предположении, что: а) дана некоторая нормальная формула длиной h букв, подлежащая реализации на предложенных модулях; б) структура модулей описывается неповторными нормальными формулами или содержит их в качестве подформул, получаемых путем настройки; в) модули используются после настройки, а число их информационных входов равно K ; г) базисные операции формул, на которые настраиваются модули, и формул, подлежащих реализации с помощью этих модулей, совпадают и подчиняются сочетательному закону; д) все модули имеют одинаковый «вес» (по сложности, габаритам, стоимости и т. д.); е) прямое x_i и инверсное \bar{x}_i значения любой входной переменной равнодоступны, поэтому реализация формулы может выполняться с точностью до типа по PN -классификации; ж) таблица настроек для каждого типа формул содержит по одному представителю; з)

ограничение на нагрузочную способность модулей не рассматривается, так как реализация в основном осуществляется в классе древовидных структур. В противном случае этот вопрос решается путем дублирования соответствующих модулей.

Отметим, что некоторые из указанных ограничений будут сняты в ходе дальнейшего изложения.

Все излагаемые ниже результаты для модулей, универсальных в классе формул в базисе $\{\&, \vee, \bar{\quad}\}$, справедливы также и для базиса $\{\&, \vee, \bar{\quad}, \oplus\}$, а также для модулей, универсальных в классе всех булевых функций.

На процесс реализации булевых формул существенное влияние оказывают функциональные свойства модулей. При этом процедура реализации главным образом зависит от того, содержатся ли в записи формул, реализуемых модулями путем настройки, инверсии и универсальны или многофункциональны используемые модули в классе неповторных формул.

Булева формула называется положительно монотонной, если в ее записи отсутствуют инверсии. В дальнейшем будем называть такую формулу монотонной.

Логические модули, структура которых описывается монотонными формулами, будем называть монотонными. Модуль, реализующий путем настройки представителей всех типов неповторных монотонных формул из K и менее букв, будем называть монотонным K -универсальным модулем. Модуль, реализующий путем настройки лишь некоторых представителей типов неповторных монотонных формул из K и менее букв, будем называть монотонным K -многофункциональным модулем. Модуль, реализующий путем настройки представителей всех типов неповторных формул из K и менее букв, среди которых имеются и немонотонные, будем называть немонотонным K -универсальным модулем. Модуль, реализующий путем настройки лишь некоторых представителей типов неповторных формул из K и менее букв, среди которых имеются и немонотонные, будем называть немонотонным K -многофункциональным модулем.

Ниже рассматриваются вопросы реализации булевых формул в базисе указанных разновидностей модулей.

Реализация монотонной неповторной формулы на монотонных K -универсальных модулях. Будем представлять реализуемую формулу и неповторные формулы, реализуемые модулем путем настройки, в виде арифметических полиномов. Такая символика позволяет отвлечься от обозначений переменных и наличия инверсий в формулах, так как она соответствует изображению формулы с точностью до типа по K -классификации. При этом предлагается сначала реализовать формулу с точностью до типа по указанной классификации, а затем — саму формулу.

Таким образом, процесс реализации следует начинать с представления исходной формулы и списка неповторных формул из K и менее букв, реализуемых модулем путем настройки, в виде арифметических полиномов.

Выбор оптимальной стратегии реализации. Покажем, что оптимальной стратегией, обеспечивающей минимальную элементную сложность при построении схемы, является такая, при которой на каждом шаге независимо от других шагов у каждого модуля задействуется максимальное число информационных входов.

Докажем следующее утверждение: если неповторная формула из h букв оптимально реализуется структурой, содержащей PN -универсальных модулей, то добавление одной буквы, соответствующей новой переменной, в любое место формулы не позволяет сократить число модулей в схеме.

Пусть формула из h букв реализуется оптимально схемой из L модулей. Добавим в формулу одну букву. Предположим, что оптимальная схема, реализующая формулу из $h+1$ букв, содержит $L-1$ модулей. Приравняем введенную букву нулю или единице. Тогда остаточная формула из h букв будет реализована схемой из $L-1$ модулей. Следовательно, исходная схема не была оптимальной, что противоречит высказанному предположению.

Так как добавление новой буквы в оптимально-реализованную формулу не приводит к упрощению схемы, то, следовательно, у каждого из модулей оптимальной схемы должно быть задействовано максимальное число информационных входов в соответствии со структурой реализуемой формулы. Поэтому стратегия реализации, при которой на некотором шаге реализуется меньшая часть формулы по сравнению с возможностями модуля, с тем чтобы на следующем шаге использовать все возможности модуля, не является оптимальной.

Таким образом, для того чтобы число модулей в схеме было минимальным, необходимо обеспечить максимальное значение числа задействованных входов у каждого из модулей в структуре на каждом

шаге реализации. При этом под шагом реализации подразумевается установка очередного модуля в схему.

Высказанные соображения позволяют принять порядок реализации схемы «от входа к выходу». При реализации неповторных формул на модулях рассматриваемого вида отдельные шаги при движении от входа схемы к ее выходу независимы в том смысле, что, принимая оптимальное решение на каждом шаге, мы не ухудшаем окончательного результата.

Дальнейшая разработка метода реализации связана с введением понятия K -отделимого фрагмента. Произвольный отрезок формулы, включающий ее концы, являющиеся буквами, и $2b$ парных скобок, где $b = 0, 1, 2, \dots$, назовем фрагментом.

В формуле из h букв заключим в скобки фрагмент из букв. Если при этом булева функция, соответствующая исходной формуле, не изменится, то назовем этот фрагмент $/$ (-отделимым).

Пример. Фрагмент $\varphi_1 = (x_1 x_2 x_3 \vee x_4) x_5$ формулы $F = (x_1 x_2 x_3 \vee x_4) x_5 x_6 x_7 \vee x_8 x_9$ является 5-отделимым, так как имеет место равенство $F = F'$, где $F' = [(x_1 x_2 x_3 \vee x_4) x_5] x_6 x_7 \vee x_8 x_9$, в то время как фрагмент

$\varphi_2 = x_5 x_6 x_7 \vee x_8 x_9$ не является отделимым, ввиду того что $F \neq F''$, где $F'' = (x_1 x_2 x_3 \vee x_4) (x_5 x_6 x_7 \vee x_8 x_9)$.

Справедливы следующие утверждения: а) произвольный фрагмент конъюнкции из K букв является K -отделимым; б) произвольный фрагмент из K букв, заключенный в парные скобки, K -отделим; в) произвольная ДНФ из K букв является K -отделимой; г) произвольная конъюнктивная нормальная форма (КНФ) из K букв является K -отделимой; д) произвольный фрагмент из K букв, являющийся конъюнкцией скобочных выражений и булевых переменных, K -отделим; е) произвольный фрагмент из K букв, являющийся дизъюнкцией конъюнкций скобочных выражений и булевых переменных, K -отделим.

Два фрагмента булевой формулы будем называть непересекающимися, если в этих фрагментах отсутствуют общие буквы. Процесс реализации булевой формулы состоит в выделении непересекающихся K -отделимых фрагментов и замене каждого из них одной новой буквой с целью уменьшения числа букв в исходной формуле.

Процедура реализации монотонной неповторной булевой формулы на монотонных K -универсальных модулях, обеспечивающая построение минимальной по сложности схемы.

1. Исходная формула и неповторные формулы, реализуемые модулем путем настройки из K и менее букв, представляются в виде арифметических полиномов.
2. Выделяются путем расстановки скобок все непересекающиеся K -отделимые фрагменты; при этом формулу рассматривают слева направо.
3. Если существует только один K -отделимый фрагмент, то он реализуется путем установки модуля, настроенного на формулу, соответствующую этому фрагменту. Реализованный фрагмент заменяется в арифметическом полиноме единицей, что приводит к формированию остаточной формулы меньшей длины.
4. Если существует несколько непересекающихся K -отделимых фрагментов, то возможны две стратегии: а) реализация образующегося после каждого шага крайнего слева K -отделимого фрагмента; б) последовательная реализация всех выделенных K -отделимых фрагментов. Выделение новых фрагментов в этом случае осуществляется лишь после реализации последнего из фрагментов, найденных на предыдущем шаге. Такая последовательность реализации позволяет уменьшить «глубину» схемы.
5. Если в исходной формуле или в рассматриваемой остаточной K -отделимые фрагменты отсутствуют, то выделяются путем расстановки скобок все непересекающиеся фрагменты, имеющие максимальный суммарный ранг K' ($K' < K$).
6. Если выделен только один K' -отделимый фрагмент, то он реализуется. После этого формируется новая остаточная формула путем замены реализованного фрагмента единицей.
7. Если существует несколько K' -отделимых фрагментов, то выделяются фрагменты, заключенные в самые глубокие скобки. При этом предполагается, что каждая «завершенная» конъюнкция произвольных фрагментов, суммарный ранг которой $r > K'$, должна быть заключена в скобки.
8. Если выделен только один K' -отделимый фрагмент, заключенный в самые глубокие скобки, то он реализуется. Этот фрагмент заменяется в арифметическом полиноме единицей, что приводит к формированию остаточной формулы.
9. Если выделено несколько K' -отделимых фрагментов, заключенных в самые глубокие скобки, то возможны две стратегии: а) реализация того из этих фрагментов, который является крайним слева; б) реализация того из фрагментов, который приводит к уменьшению «глубины» схемы. Для этого буквам,

образующим формулу, присваиваются веса: буквам исходной формулы - вес, равный 0, а каждой новой букве, соответствующей реализованному фрагменту, - вес, равный максимальному весу одной из букв, входящих в этот фрагмент, увеличенному на единицу. При этом реализуется тот фрагмент, у которого максимальный вес минимален. Реализация исходной формулы завершается, если число букв в остаточной равно единице. Пример. Реализовать формулу

$y = [(x_1 \vee x_2)x_3 \vee x_4x_5x_6 \vee x_7]x_8x_9 \equiv [(1+1)1+3+1]2$ на 3-универсальных модулях, структура которых описывается функцией $F = (z_1 \vee z_2 \vee z_3)z_4 \vee z_1z_2z_3$.

Каждый из этих модулей путем настройки реализует следующих четырех представителей типов неповторных формул из трех букв: при $z = 0$ $F = z_1z_2z_3 = 3$; при $z_4 = 1$ $F = z_1 \vee z_2 \vee z_3 \equiv 1+1+1$; при $z_3 = 0$ $F = (z_1 \vee z_2)z_4 \equiv (1+1)1$; при $z_3 = 1$ $F = z_1z_2 \vee z_4 \equiv 2+1$.

Процедура реализации:

- | | |
|---------------------------------|-----------------------------------|
| 1. Выделяем фрагмент $(1+1)1$. | Остаток - $y' \equiv (1+3+1)2$. |
| 2. Выделяем фрагмент 3. | Остаток - $y'' \equiv (1+1+1)2$. |
| 3. Выделяем фрагмент $1+1+1$. | Остаток - $y''' \equiv 3$. |
| 4. Выделяем фрагмент 3. | Остаток - $y^{IV} \equiv 1$. |

Реализация формулы завершена. Использовано четыре 3-универсальных модуля.

Комментарий к процедуре:

1. В изложенной процедуре реализация каждой остаточной формулы начинается с попытки выделить все X-отделимые фрагменты, и лишь в том случае, когда это невозможно, осуществляется переход к выделению фрагментов, имеющих максимальный суммарный ранг $K'(K' < K)$. Такая процедура обеспечивает реализацию на каждом шаге фрагмента максимальном длине и позволяет реализовать исходную формулу схемой с минимальным числом модулей.

2. Рассмотрим примеры, демонстрирующие, что если несколько отделимых фрагментов имеют максимальный суммарный ранг $K'(K' < K)$, то реализация должна начинаться с фрагмента, заключенного в самые глубокие скобки. Покажем, что, если выполнять реализацию, не пользуясь этим правилом, можно получить неоптимальную схему.

Пример. Реализовать формулу $y = (x_1 \vee x_2 \vee x_3)(x_4x_5x_6 \vee x_7x_8x_9)$ на монотонных 4-универсальных модулях.

Так как 4-отделимые фрагменты отсутствуют, то построение схемы начинается с выделения 3-отделимых фрагментов: $y = (x_1 \vee x_2 \vee x_3)[(x_4x_5x_6) \vee (x_7x_8x_9)]$.

Таким образом, исходная формула содержит три 3-отделимых фрагмента. Если начать реализацию с крайнего слева фрагмента $\varphi_1 = x_1 \vee x_2 \vee x_3$, то схема будет содержать четыре модуля. Начиная реализацию с фрагмента $\varphi_2 = x_4x_5x_6$, заключенного в самые глубокие скобки, получим схему, содержащую три модуля.

Приведем пример, демонстрирующий, что если в формуле несколько отделимых фрагментов, имеющих максимальный суммарный ранг $K'(K' < K)$ и заключенных в самые глубокие скобки, то реализация формулы должна начинаться с того из этих фрагментов, у которого максимальный вес минимален. Указанный порядок реализации позволяет уменьшить «глубину» схемы.

Пример. Реализовать формулу $y = (x_1 \vee x_2x_3x_4x_5)(x_6 \vee x_7 \vee x_8) \vee x_9(x_{10} \vee x_{11} \vee x_{12})$ на монотонных 5-универсальных модулях.

Присвоим всем буквам исходной формулы вес, равный нулю. На первом шаге осуществляется реализация 5-отделимых фрагментов $\varphi_1 = x_1 \vee x_2x_3x_4x_5$. При этом остаточная формула имеет вид $y' = \varphi_1(x_6 \vee x_7 \vee x_8) \vee x_9(x_{10} \vee x_{11} \vee x_{12})$.

Присвоим букве φ_1 вес, равный 1. Так как формула содержит два 4-отделимых фрагмента, заключенных в скобки одинаковой глубины, то выбор фрагмента осуществим на основе изложенного выше правила. Для первого фрагмента максимальный вес равен 1, а для второго - 0. Поэтому первоначально реализуем фрагмент $\varphi_2 = x_9(x_{10} \vee x_{11} \vee x_{12})$. При этом остаточная формула имеет вид $y'' = \varphi_1(x_6 \vee x_7 \vee x_8) \vee \varphi_2$ и реализуется на последнем шаге. Таким образом, оптимальная схема содержит три модуля и имеет два яруса. При реализации на втором шаге крайнего слева фрагмента $\varphi_3 = \varphi_1(x_6 \vee x_7 \vee x_8)$ формула будет реализована трехъярусной схемой, также содержащей три модуля.

Реализация монотонной повторной формулы на монотонных K -универсальных модулях. Процедура реализации монотонной повторной формулы на монотонных K -универсальных модулях практически не отличается от рассмотренной выше. Реализуемая повторная формула заменяется бесповторной формулой, содержащей то же число букв h и имеющей тот же арифметический полином. Полученная бесповторная формула реализуется на основе процедуры, изложенной выше.

В построенной схеме отождествляются входы в соответствии с заданной повторной формулой. Так как число букв и структура повторной формулы не отличаются от аналогичных характеристик соответствующей бесповторной формулы, то число модулей, требующихся для их реализации, одинаково. Отметим, что при этом получается вариант реализации в виде древовидной структуры, который может отличаться от минимального.

Реализация немонотонной формулы в базисе монотонных K -универсальных модулей. Реализация немонотонной нормальной формулы на монотонных $/(-$ -универсальных модулях при равной доступности прямых и инверсных значений входных переменных практически не отличается от рассмотренной выше. Исходная формула записывается в виде арифметического полинома, к которому применяется изложенная выше процедура реализации. На входы построенной схемы, которым в формуле соответствуют переменные с инверсией, сигналы подаются с инверсных выходов *III*.

Если исходная формула не является нормальной, т. е. в ней имеются инверсии над группами переменных, то первоначально формулу необходимо привести к нормальному виду, после чего к ней применяется изложенная процедура. При приведении формулы к нормальному виду обычно многократно применяется правило де Моргана, использование которого связано с громоздкими преобразованиями. С целью упрощения этой операции предлагается следующий алгоритм.

Преобразование булевой формулы к нормальному виду. Подсчитывается число инверсий над каждой буквой и каждым знаком операции в формуле. При четном числе инверсий соответствующее обозначение не изменяется. При нечетном - меняется на дополнительное.

Отметим, что в результате указанного преобразования число букв в формуле не изменяется и поэтому оценить сложность схемы можно непосредственно по заданной формуле, не приводя ее к нормальному виду.

Выше был рассмотрен случай реализации при условии, что прямые и инверсные выходы *III* равнодоступны. В случае, когда доступны лишь прямые выходы *III*, сначала реализуется соответствующая положительно монотонная формула, а после этого на «инверсных» входах устанавливаются инверторы.

Реализация булевой формулы на немонотонных K -универсальных модулях. Выше было показано, что не монотонность исходной формулы не приводит к существенному изменению процедуры, предложенной для реализации монотонной формулы. Покажем, что появление инверсий в списке формул, реализуемых модулем при соответствующей настройке, требует значительного дополнения этой процедуры.

Разработка этой процедуры основывается на свойстве универсальности используемых модулей, обеспечивающем в таблице настроек для каждой из формул наличие соответствующей ей дополнительной формулы, по крайней мере с точностью до типа по PN -классификации.

Сформулируем ряд определений, которые будут использованы в дальнейшем.

Формула $f^*(x_1, \dots, x_n)$ называется дополнительной для формулы $f(x_1, \dots, x_n)$, если для соответствующих им булевых функций справедливо равенство

$$f^*(x_1, \dots, x_n) = \overline{f(x_1, \dots, x_n)} \quad (4-1)$$

Формула, дополнительная к бесповторной формуле, также является бесповторной и содержит то же число букв. Структурной будем называть схему, отражающую число модулей, связи между ними и формулы, реализуемые этими модулями при настройке их с точностью до типа по NPN -классификации. Вход модуля, для которого в реализуемой этим модулем формуле соответствующая буква инверсна, будем называть инверсным.

Процедура реализации булевой формулы на немонотонных K -универсальных модулях. Процесс реализации формулы в этом случае состоит из двух этапов. На первом этапе без учета инверсий в заданной формуле и формулах, реализуемых модулем при соответствующей настройке, с помощью процедуры, изложенной выше для монотонных K -универсальных модулей, от входа к выходу строится структурная схема.

На втором этапе структурная схема преобразуется от выхода к входу с учетом инверсий, входящих в описание формул базиса, с целью определения конкретных формул, реализуемых модулями в схеме. Окончательная настройка модулей определяется по правилу: если рассматриваемый модуль является выходным или от выхода рассматриваемого модуля до выхода схемы сигнал «проходит» четное число инверсий, то этот модуль должен быть настроен на реализацию формулы, принадлежащей одному *PN*-типу с положительно монотонной формулой, приписанной модулю на первом этапе; если от выхода рассматриваемого модуля до выхода схемы сигнал «проходит» нечетное число инверсий, то этот модуль должен быть настроен на реализацию формулы, принадлежащей к одному типу с формулой, дополнительной к положительно-монотонной формуле, приписанной модулю на первом этапе. Указанное правило распространяется и на входы схемы: если прямые и инверсные выходы *III* равнодоступны, то тот вход схемы, от которого сигнал до ее выхода «проходит» четное число инверсий, получает его с выхода источника, соответствующего инверсности буквы в реализуемой формуле, а тот вход схемы, от которого сигнал до ее выхода «проходит» нечетное число инверсий, - с выхода источника, противоположного инверсности буквы в формуле.

В случае, когда доступны лишь прямые выходы *III*, на тот вход схемы, от которого сигнал до ее выхода «проходит» четное (нечетное) число инверсий, размещают инверторы, если соответствующая буква в формуле инверсна (безынверсна).

Выбор выходов *III* или входов, на которых должны быть установлены инверторы, может быть осуществлен также путем сравнения заданной формулы и формулы нормального вида, описывающей структуру схемы после выполнения второго этапа.

Все сказанное выше относилось к реализации неповторной формулы. В случае, если реализуемая формула повторна, построение схемы выполняется аналогично, а после завершения синтеза осуществляется отождествление входов.

Комментарий к процедуре: 1) в случае, если модуль при соответствующей настройке может реализовать несколько различных формул, принадлежащих одному *PN*-типу, то выбор одной из них связан с перебором. Для устранения этого вида перебора может быть введено какое-либо правило, состоящее, например, в том, что выбирается первая из списка этих формул; 2) в случае, если в составе дизъюнкции или конъюнкции в формуле, реализуемой последующим модулем при его настройке, часть букв инверсна, то возможен перебор на втором этапе реализации, при подключении предыдущего модуля. Для устранения этого вида перебора также может быть введено соответствующее правило выбора.

При равнодоступности прямых и инверсных выходов *III* и одинаковой сложности всех используемых модулей вне зависимости от объема перебора сложность получаемых схем не изменяется. Если доступны лишь прямые выходы *III*, то в зависимости от объема перебора изменяется число инверторов на входе схемы.

Пример. Реализовать формулу

$$f = [\bar{z}_1 \bar{z}_2 \vee z_3 z_4 z_5 z_6 \vee (z_7 z_8 z_9 \vee \bar{z}_{10} \vee \bar{z}_{11}) z_{12} z_{13}] (\bar{z}_{14} \vee \bar{z}_{15})$$

на немонотонных 3-универсальных модулях, реализующих путем настройки следующие формулы:

$$\varphi_1 = \bar{x}_1 x_2 \vee \bar{x}_3; \varphi_2 = (\bar{x}_1 \vee \bar{x}_2) \bar{x}_3; \varphi_3 = \bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3; \varphi_4 = x_1 x_2 x_3,$$

предполагая, что прямые и инверсные выходы *III* равнодоступны. Запишем арифметические полиномы рассматриваемых формул:

$$f \equiv [2 + 4 + (3 + 1 + 1)2](1 + 1); \\ \varphi_1 \equiv 2 + 1; \varphi_2 \equiv (1 + 1)1; \varphi_3 \equiv 1 + 1 + 1; \varphi_4 \equiv 3;$$

Первый этап (рис. 4-1,а):

1. Выделяем фрагмент 3 и устанавливаем первый элемент. Остаток

$$f_1 \equiv [2 + 2 + (3 + 1 + 1)2](1 + 1).$$

2. Выделяем фрагмент 3 и устанавливаем второй элемент. Остаток

$$f_2 \equiv [2 + 2 + (1 + 1 + 1)2](1 + 1).$$

3. Выделяем фрагмент 1+1+1 и устанавливаем третий элемент. Остаток

$$f_3 \equiv (2 + 2 + 3)(1 + 1).$$

4. Выделяем фрагмент 3 и устанавливаем четвертый элемент. Остаток

$$f_4 \equiv (2 + 2 + 1)(1 + 1).$$

5. Выделяем фрагмент 2+1 и устанавливаем пятый элемент. Остаток

$$f_5 \equiv (2 + 1)(1 + 1).$$

6. Выделяем фрагмент 2+1 и устанавливаем шестой элемент. Остаток

$$f_6 \equiv 1(1 + 1).$$

7. Выделяем фрагмент 1(1+1) и устанавливаем седьмой элемент. Остаток

$$f_7 \equiv 1.$$

Второй этап:

7. Настройка седьмого модуля не изменяется:

$$\overline{x_3}(\overline{x_2} \vee \overline{x_1}).$$

6. Число инверсий до выхода схемы нечетно. Настройка шестого модуля изменяется на дополнительную: $(\overline{x_1} \vee \overline{x_2})\overline{x_3}$.

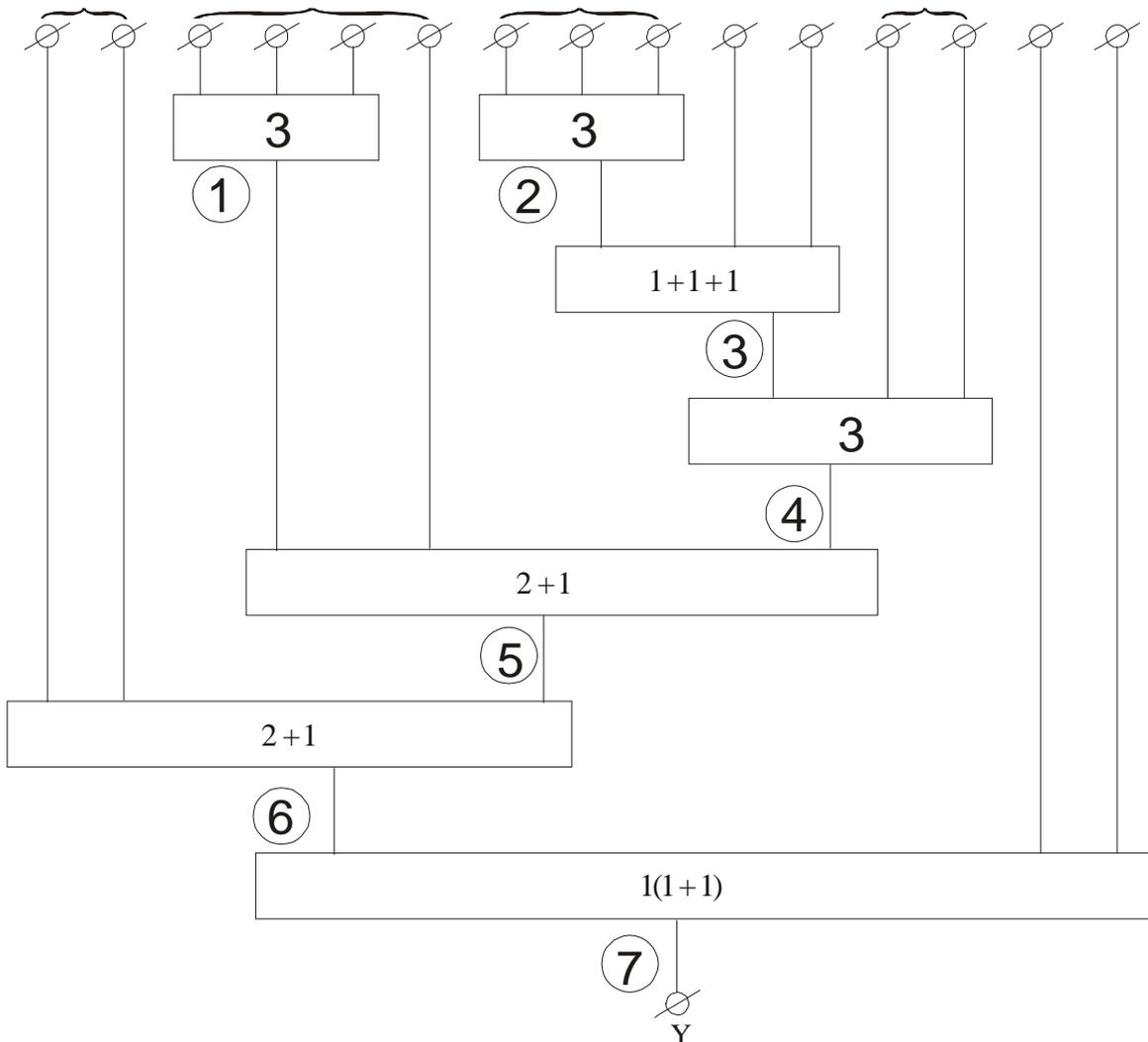
5. Число инверсий до выхода схемы четно. Настройка пятого модуля не изменяется: $\overline{\overline{x_1} \overline{x_2}} \vee \overline{x_3}$.

4. Число инверсий до выхода схемы нечетно. Настройка четвертого модуля не изменяется: $\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$.

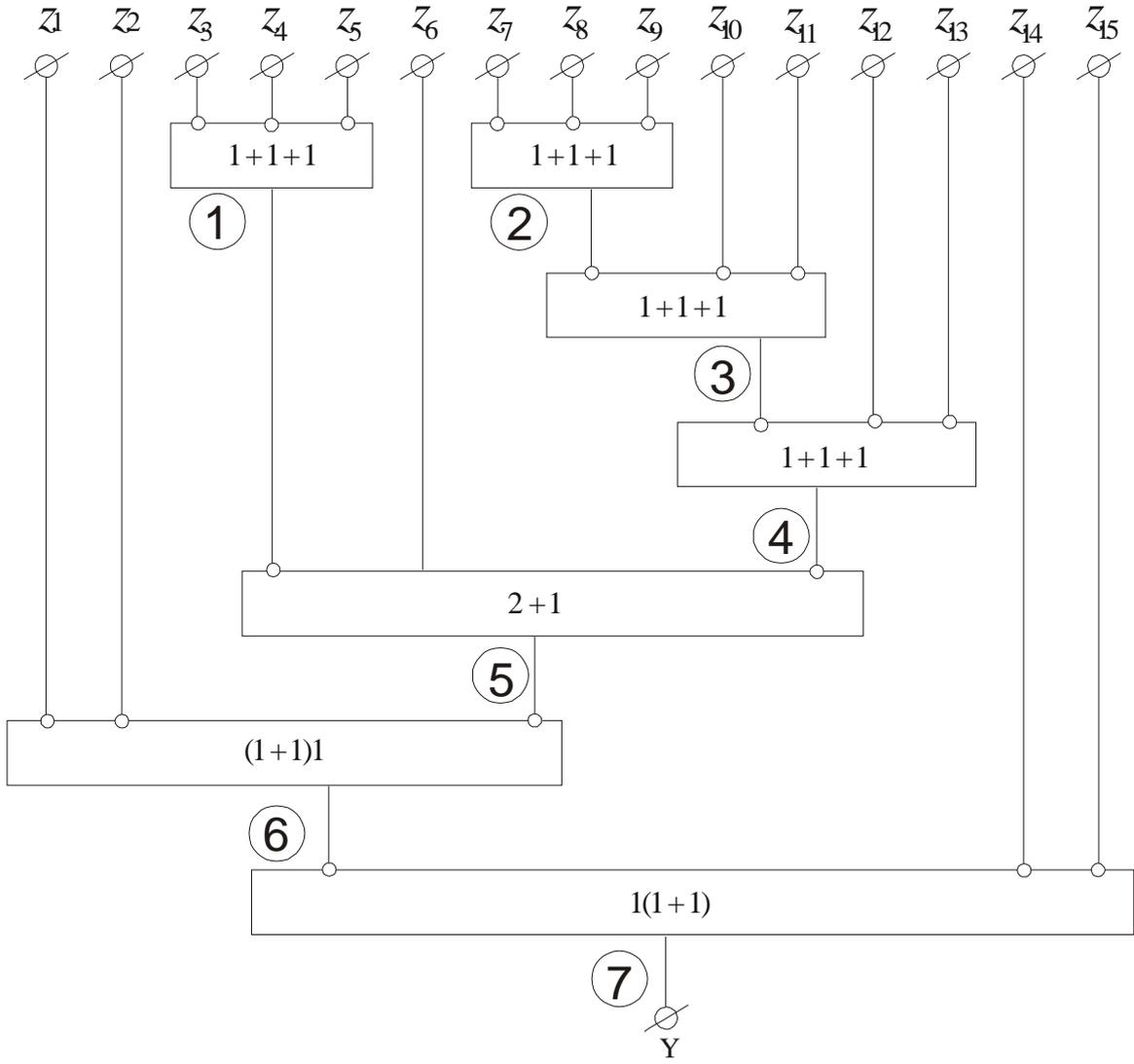
3. Число инверсий до выхода схемы нечетно. Настройка третьего модуля не изменяется: $\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$.

2. Число инверсий до выхода схемы нечетно. Настройка четвертого модуля не изменяется: $\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$.

A) [2 + 4 + (3 + 1 + 1) 2] (1+1)



Б)



B)

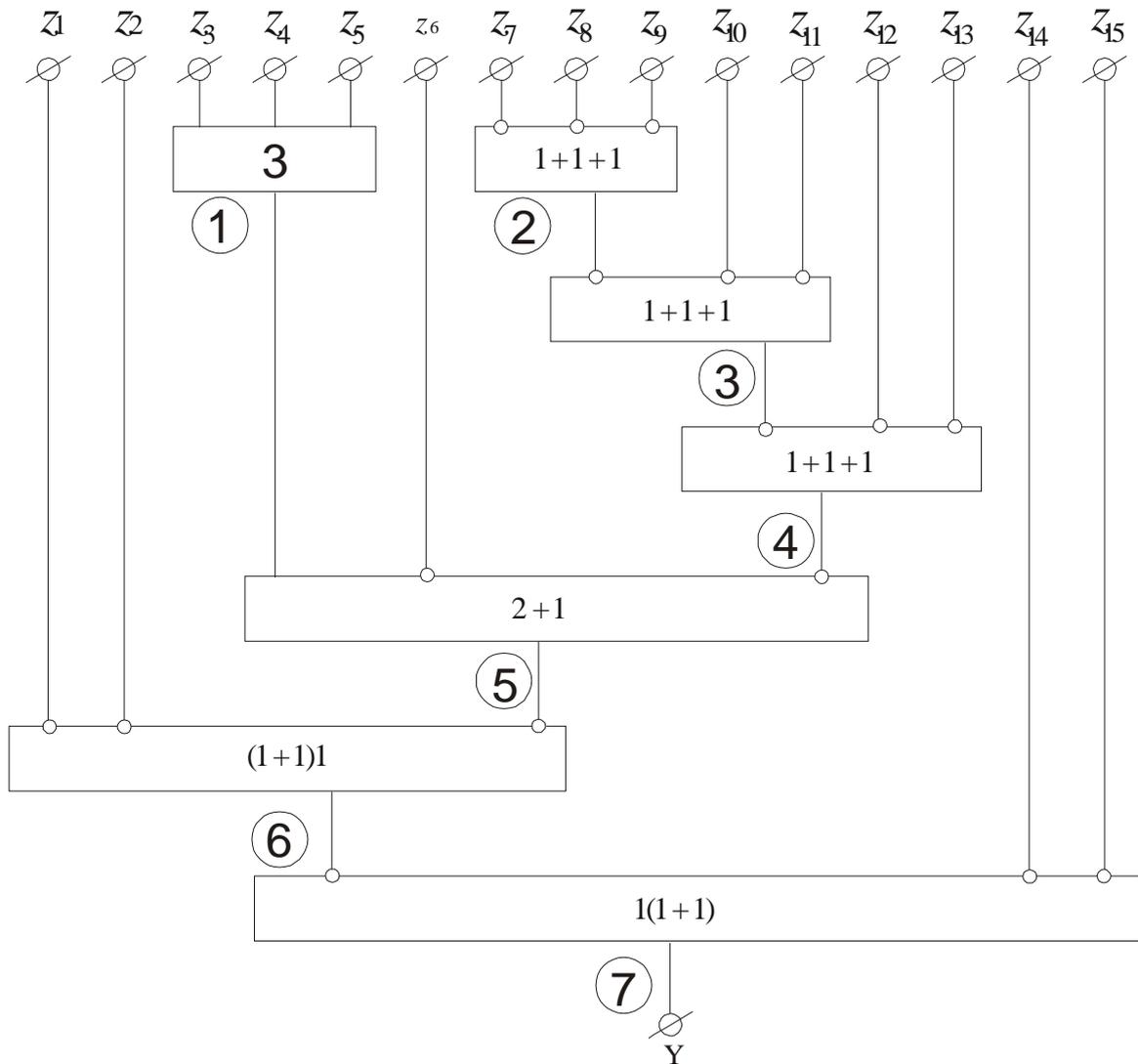


Рис. 4-1. Реализация булевой формулы на немонотонных универсальных модулях: а - построение структурной схемы; б - первый способ корректировки структурной в - второй способ корректировки структурной схемы

1. Здесь возможны два варианта: а) при подключении первого модуля к инверсному входу пятого модуля число инверсий от выхода первого модуля до выхода схемы нечетно, и поэтому его настройка не изменяется: $\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3$ (рис. 4-1, б). Определим формулу, реализуемую построенной схемой:

$$f = (\bar{z}_1 \vee \bar{z}_2)(\bar{z}_3 \vee \bar{z}_4 \vee \bar{z}_5)z_6 \vee \bar{z}_7 \vee \bar{z}_8 \vee \bar{z}_9 \vee \bar{z}_{10} \vee \bar{z}_{11} \vee \bar{z}_{12} \vee \bar{z}_{13}(\bar{z}_{14} \vee \bar{z}_{15}) =$$

$$= [z_1 z_2 \vee z_3 z_4 z_5 z_6 \vee (z_7 z_8 z_9 \vee \bar{z}_{10} \vee \bar{z}_{11})z_{12} z_{13}](\bar{z}_{14} \vee \bar{z}_{15})$$

Из сопоставления полученной формулы с заданной следует, что на первый и второй входы схемы сигналы должны поступать либо через инверторы, либо с инверсных выходов *III*;

б) при подключении первого модуля к безынверсному входу пятого модуля число инверсий от выхода первого модуля до выхода схемы четно, и поэтому его настройка не изменяется: $x_1 x_2 x_3$ (рис. 4-1, в).

Определим формулу, реализуемую построенной схемой:

$$f = (\bar{z}_1 \vee \bar{z}_2)z_3 z_4 z_5 z_6 \vee \bar{z}_7 \vee \bar{z}_8 \vee \bar{z}_9 \vee \bar{z}_{10} \vee \bar{z}_{11} \vee \bar{z}_{12} \vee \bar{z}_{13}(\bar{z}_{14} \vee \bar{z}_{15}) =$$

$$= [z_1 z_2 \vee z_3 z_4 z_5 z_6 \vee (z_7 z_8 z_9 \vee \bar{z}_{10} \vee \bar{z}_{11})z_{12} z_{13}](\bar{z}_{14} \vee \bar{z}_{15})$$

Из сопоставления полученной формулы с заданной следует, что на первый, второй и шестой входы схемы сигналы должны поступать либо через инверторы, либо с инверсных выходов *III*.

Реализация булевых формул на немонотонных K -универсальных модулях, реализующих при настройке всех представителей P -типов бесповторных формул. Пусть заданная формула реализуется на модулях, универсальных в классе всех представителей P -типов бесповторных формул в базисе $\{\&, \vee, \bar{}\}$. При $K = 2$ такой модуль реализует представителей шести P -типов бесповторных формул из двух букв: $y_1 = \bar{x}_1 \bar{x}_2$; $y_2 = \bar{x}_1 x_2$; $y_3 = x_1 x_2$; $y_4 = \bar{x}_1 \vee \bar{x}_2$; $y_5 = \bar{x}_1 \vee x_2$; $y_6 = x_1 \vee x_2$, а при $K = 3$ модуль реализует представителей 20 P -типов бесповторных формул из трех букв.

Для таких модулей характерно, что: а) при их использовании процесс реализации выполняется в один этап от «входа к выходу»; б) даже в случае неравной доступности выходов III нет необходимости в установке инверторов на входе схемы.

Процесс реализации при использовании таких модулей сводится к последовательному выделению K -отделимых фрагментов, каждый из которых реализуется с точностью до требуемой подформулы.

Пример. Реализовать формулу $y = (\bar{x}_1 \vee \bar{x}_2)x_3 \vee x_4 \bar{x}_5$ в базисе 3-универсальных модулей рассматриваемого типа.

1. Выделяем фрагмент $\varphi_1 = (\bar{x}_1 \vee \bar{x}_2)x_3$ и реализуем его.

Остаток $y' = \varphi_1 \vee x_4 \bar{x}_5$

2. Выделяем фрагмент $\varphi_2 = \varphi_1 \vee x_4 \bar{x}_5$ и реализуем его. Построение схемы закончено (рис. 4-2).

Реализация булевых формул на монотонных K_1 -многофункциональных модулях. Процедура реализации булевой формулы монотонных K_1 -многофункциональных модулях во многом близка к процедуре реализации на монотонных K_1 -универсальных модулях. Однако, если при использовании K_1 -универсальных модулей любой K_1 -отделимый фрагмент может быть реализован Виним модулем, то при применении K_1 -многофункциональных модулей, универсальных в классе формул из K букв, одним модулем может быть реализован лишь любой K -отделимый фрагмент ($K \leq K_1$), а из всех K^* -отделимых фрагментов ($K < K^* \leq K_1$) одним модулем могут быть реализованы лишь те из них, представители типов которых входят в таблицу настроек.

Основная идея излагаемого подхода, как и в случае использования универсальных модулей, состоит в том, чтобы на каждом шаге выделялся и реализовался отделимый фрагмент наибольшей длины, совпадающий с формулами, приведенными в таблице настроек. Отличие излагаемого подхода от реализации в базисе универсальных модулей состоит в том, что при $K_1 > K$ перед реализацией выделенного фрагмента необходимо выполнить сравнение его с формулами, входящими в таблицу настроек. Если в таблице настроек формула, соответствующая выделенному фрагменту, отсутствует, то, следовательно, этот фрагмент не может быть реализован. Поэтому некоторые шаги процедуры в этом случае могут быть нерезультативными, в то время как при реализации на универсальных модулях каждый из шагов результативен.

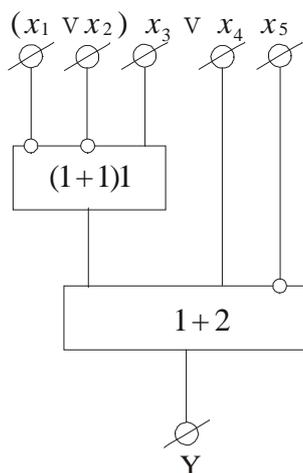


Рис. 4-2. Реализация булевой формулы на немонотонных универсальных модулях, реализующих всех представителей P -типов бесповторных формул

Если некоторый шаг процедуры, связанный с выделением фрагмента длиной, большей K , заканчивается безрезультатно, то этот факт еще не свидетельствует о том, что фрагменты этой длины не могут быть выделены. В этом случае должно быть выполнено новое выделение фрагментов этой же

длины, которое может закончиться реализацией выделенных фрагментов. Таким образом, при реализации формул на многофункциональных модулях возможна разновидность перебора, которая отсутствовала при использовании универсальных модулей.

Параметр K_I для многофункционального модуля показывает, от какого максимального числа букв модуль при настройке реализует хотя бы одну неповторную формулу.

Для того чтобы произвольная формула могла быть реализована на K_I -многофункциональных модулях при условии равнодоступности прямых и инверсных выходов III , этот модуль должен быть по крайней мере 2-универсальным, т. е. он должен быть способен при настройке реализовать формулы 2 и $1 + 1$.

Реализация булевой формулы в базисе немонотонных K_I -многофункциональных модулей. В силу того что рассматриваемые модули немонотонны, процесс реализации должен состоять из-за двух этапов, а ввиду того что модуль K_I -многофункционален, процедура выполнения первого этапа должна совпадать с рассмотренной выше.

Однако специфика этого случая состоит в том, что если некоторая формула, использованная на первом этапе синтеза, на втором этапе должна быть заменена на дополнительную, а дополнительная формула в таблице настроек отсутствует, то второй этап реализации не завершается и необходимо вновь переходить к первому этапу, строя новую структурную схему.

Таким образом, если для модулей, рассмотренных выше, удавалось добиться оптимальной реализации, пользуясь процедурой, в которой шаги независимы, то в данном случае нарушается принцип независимости отдельных шагов.

Устранить указанный недостаток можно в том случае, если выбрать такие модули, у которых в таблице настроек для каждой формулы имеется дополнительная, по крайней мере с точностью до PN -типа. Если в таблице настроек модуля не каждая из формул имеет дополнительную, то использовать все логические возможности модулей удастся, если условно ввести в таблицу настроек для каждой из формул дополнительную и применять их на первом этапе реализации наравне с реально существующими.

При этом, если в ходе первого этапа требуется использовать условно введенную формулу, то в случае, когда на втором этапе она должна быть изменена на дополнительную, процесс реализации сходится, а в случае, когда изменение настройки не требуется, процесс не сходится.

Процесс реализации упрощается и обязательно сходится при ограничении таблицы настроек и использовании только тех формул, которые имеют дополнительные, что, естественно, приводит к некоторому снижению логической эффективности модулей.

Второй этап может быть завершен, если на выходе модуля, который не позволяет закончить реализацию, установить инвертор; однако этот путь не является рекомендуемым, так как добавляется еще один тип элементов, вес которых отличен от веса модулей.

4-2. ОЦЕНКИ СЛОЖНОСТИ РЕАЛИЗАЦИИ

Пусть задана монотонная неповторная формула из h букв и монотонные K -универсальные модули. Требуется определить, нижнюю и верхнюю оценки числа модулей $L(h; K)$, необходимых для реализации заданной формулы.

В работе [5] показано, что искомые оценки имеют следующий вид:

$$\left\lceil \frac{h-1}{K-1} \right\rceil \leq L(h; K) \leq \left\lceil \frac{2(h-1)}{K} \right\rceil. \quad (4-2)$$

Анализ оценок сложности. Приведенные оценки отражают не свойства того или иного метода реализации (при «хорошем» методе - нижняя, а при «плохом» - верхняя), а объективные свойства реализуемых формул и используемых модулей.

Наличие интервала, задаваемого верхней и нижней оценками, при фиксированных h и K определяется видом реализуемой формулы, так как при $h = \text{const}$ существует $S(h)$ различных PN -типов формул, некоторые из которых имеют структуру, «изоморфную» структуре используемого модуля, что позволяет выполнять реализацию с числом элементов, близким или равным нижней оценке, а для других формул этот «изоморфизм» отсутствует и реализация выполняется с числом элементов, равным или близким верхней оценке.

Рассмотрим отношение

$$\frac{L_B}{L_H} \approx 2 \times \frac{K-1}{K} \ll 2. \quad (4-3)$$

Из этого отношения следует, что при $K = \text{const}$ сложность наиболее «сложной» формулы из h букв, определяемая числом модулей, требующихся для ее реализации, не превосходит более чем в два раза сложности наиболее «простой» формулы из того же числа букв.

Продemonстрируем высказанное утверждение на примерах. Число модулей, необходимое для реализации формул из пяти букв на 3-универсальных модулях, удовлетворяет неравенству

$$2 = \left\lceil \frac{5-1}{3-1} \right\rceil \leq L(5;3) \leq \left\lceil \frac{2(5-1)}{3} \right\rceil = 3.$$

При этом из представителей 24 РЛ/-типов неповторных формул в базисе $\{\&, \vee, \bar{}\}$ 22 требуют двух модулей, а два «неизоморфных» представителя $y_1 = (x_1 \vee x_2)(x_3 \vee x_4) \vee x_5$ и $y_2 = (x_1 x_2 \vee x_3 x_4) x_5$ - трех модулей.

Все представители 66 PN-типов таких формул из шести букв реализуются по нижней оценке, несмотря на то что верхняя оценка равна четырем:

$$3 = \left\lceil \frac{6-1}{3-1} \right\rceil \leq L(6;3) \leq \left\lceil \frac{2(6-1)}{3} \right\rceil = 4.$$

Из представителей 180 PN-типов таких формул из семи букв 24 реализуются по верхней оценке, а остальные - по нижней:

$$3 = \left\lceil \frac{7-1}{3-1} \right\rceil \leq L(7;3) \leq \left\lceil \frac{2(8-1)}{3} \right\rceil = 5.$$

Из представителей 522 PN -типов формул из восьми букв только четыре реализуются по верхней оценке:

$$4 = \left\lceil \frac{8-1}{3-1} \right\rceil \leq L(8;3) \leq \left\lceil \frac{2(8-1)}{3} \right\rceil = 5.$$

Необходимо отметить, что одна и та же формула может быть «сложной» при использовании одних модулей (требовать для своей реализации числа модулей, совпадающего с верхней оценкой) и «простой» при ее реализации на других модулях (требовать числа модулей, совпадающего с нижней оценкой). Например, формула, которой соответствует полином

$$F \equiv [(4+4)(1+1+1) + (4+4)(1+1+1)](1+1+1),$$

является «сложной» при $K = 7$ и «простой» при $K = 8$.

В табл. 4-1 приведены значения числа букв Л, для которых могут быть найдены формулы, требующие для своей реализации числа модулей, определяемого по верхней оценке.

Таблица 4-1

Значения числа букв h в формулах, реализуемых по верхней оценке

Универсальность модуля К	Число модулей L										
	1	2	3	4	5	6	7	8	9	10	11
2	2	3	4	5	6	7	8	9	10	11	12
3	2	4	5	7	8	10	11	13	14	16	17
4	3	5	7	9	11	13	15	17	19	21	23
5	3	6	8	11	13	16	18	21	23	26	28
6	4	7	10	13	16	19	22	25	28	31	34
7	4	8	11	15	18	22	25	29	32	36	39
8	5	9	13	17	21	25	29	33	37	41	45
9	5	10	14	19	23	28	32	37	41	46	50

10	6	11	16	21	26	31	36	41	46	51	56
11	6	12	17	23	28	34	39	25	50	56	61

Приведем пример формулы, реализуемой по верхней оценке. Такая формула для $K = 5, h=13, L= 5$ (табл. 4-1) имеет следующий вид:

$$y = [(x_1 x_2 x_3 \vee x_4 x_5 x_6)(x_7 \vee x_8) \vee x_9 x_{10} x_{11}](x_{12} \vee x_{13}).$$

Оценки сложности реализации в базисе K_I -многофункциональных модулей. Определим оценки сложности предстоящей реализации булевой формулы из h букв на K_I -многофункциональных модулях, универсальных в классе формул из K букв ($K \geq 2$).

Нижняя оценка. Минимальное число модулей для реализации заданной формулы из h букв требуется в том случае, когда у каждого модуля используются все K_I информационных входов; при этом

$$L_{\min}(h; K; K_I) = \left\lceil \frac{h-1}{K_I-1} \right\rceil. \quad (4-4)$$

Верхняя оценка. Если K_I -многофункциональный модуль универсален в классе неповторных формул из K букв, то этот модуль допускает использование не худшее, чем соответствующий K -универсальный модуль, поэтому. Таким образом, для K_I -многофункциональных модулей справедливо неравенство

$$\left\lceil \frac{h-1}{K_I-1} \right\rceil \leq L(h; K; K_I) \leq \left\lceil \frac{2(h-1)}{K} \right\rceil. \quad (4-6)$$

Очевидно, что чем ближе значения K и K_I у многофункционального модуля, тем меньше различие между верхней и нижней [оценками сложности предстоящей реализации. При $K_I = K$ разность между верхней и нижней оценками минимальна. Следовательно, чем ближе многофункциональный модуль к универсальному, тем больше точность предсказания результатов реализации.

Оценки сложности реализации булевых формул в базисе настраиваемых модулей, универсальных в классе ДНФ. В § 2-2 отмечалось, что для многих классов управляющих логических устройств характерно, что их алгоритмы описываются формулами, заданными в ДНФ. С другой стороны, модули, универсальные в классе ДНФ из K букв, обладают более простой внутренней структурой и меньшим числом внешних выводов по сравнению с модулями, универсальными в классе произвольных формул в базисе $\{\&, \vee, \bar{}\}$ из K букв. Кроме того, такие модули могут быть использованы также и для реализации булевых функций, заданных в виде произвольных скобочных формул в базисе $\{\&, \vee, \bar{}\}$, причем процедура реализации в этом базисе более проста по сравнению с соответствующей процедурой использования K -универсальных модулей.

Приведем без доказательства оценки сложности реализации формул в базисе рассматриваемых модулей.

1. *Оценки сложности реализации произвольной ДНФ из A' букв на модулях, универсальных в классе ДНФ из K' букв:*

$$\left\lceil \frac{h'-1}{K'-1} \right\rceil \leq L \leq \left\lceil \frac{h'-1}{\left\lceil \frac{K'}{2} + 1 \right\rceil} \right\rceil, \quad (4-7)$$

где $\lceil \rceil$ -символ округления до ближайшего меньшего целого числа.

2. *Оценки сложности реализации системы формул, представленных в ДНФ, на модулях, универсальных в классе ДНФ из K' букв.*

Пусть задана система из N формул y_1, y_2, \dots, y_N , представленных в ДНФ, число букв в правых частях

которых соответственно равно h'_1, h'_2, \dots, h'_N . Введем обозначение $\sum_{i=1}^N h'_i = H'$. Число модулей,

требуемых для реализации этой системы формул, удовлетворяет неравенству

$$\left\lceil \frac{H' - N}{K' - 1} \right\rceil \leq L \leq \left\lceil \frac{H' - N}{\left\lceil \frac{K'}{2} + 1 \right\rceil} \right\rceil + N - 1. \quad (4-8)$$

3. Оценки сложности реализации произвольной формулы в базисе $\{\&, \vee, \bar{}\}$ из h букв на модулях, универсальных в классе ДНФ из K' букв.

Произвольная скобочная формула в базисе $\{\&, \vee, \bar{}\}$ из h букв, число скобок в которой равно c , может быть реализована с помощью рассматриваемых модулей только при представлении ее в виде системы из

$$N = \frac{c}{2} + 1 \quad (4-9)$$

ДНФ, суммарное число букв в правых частях которых

$$H' = h + \frac{c}{2}. \quad (4-10)$$

Подставляя найденные значения N и H' в неравенство (4-8), получим соотношение

$$\left\lceil \frac{h-1}{K'-1} \right\rceil \leq L \leq \left\lceil \frac{h-1}{\left\lceil \frac{K'}{2} + 1 \right\rceil} \right\rceil + \frac{c}{2}. \quad (4-11)$$

4. Оценки сложности реализации системы произвольных формул в базисе $\{\&, \vee, \bar{}\}$ из h букв на модулях, универсальных в классе ДНФ из K' букв.

Пусть задана система из N формул y_1, y_2, \dots, y_N , число букв и скобок в правых частях которых

соответственно равно h_1, h_2, \dots, h_N и c_1, c_2, \dots, c_N . Введем обозначения $\sum_{i=1}^N h_i = H; \sum_{i=1}^N c_i = C$.

В этом случае справедливо соотношение

$$\left\lceil \frac{H - N}{K' - 1} \right\rceil \leq L \leq \left\lceil \frac{H - N}{\left\lceil \frac{K'}{2} + 1 \right\rceil} \right\rceil + \frac{C}{2} + N - 1. \quad (4-12)$$

Оценки сложности реализации булевых формул в базисе микросборок. Пусть имеется микросборка, содержащая набор из N не связанных между собой модулей, универсальных в классе формул из K_N, K_{N-1}, \dots, K_1 букв. Требуется определить оценки сложности реализации булевой формулы из h букв, выраженные в числе этих микросборок.

Нижняя оценка. Максимальное число букв, для которого может быть реализована формула в базисе рассматриваемой микросборки, определяется из соотношения

$$K = \left(\sum_{i=1}^N K_i \right) - (N - 1). \quad (4-13)$$

При этом

$$L_{\text{ниж}} = \left\lceil \frac{h-1}{K-1} \right\rceil = \left\lceil \frac{h-1}{\left(\sum_{i=1}^N K_i \right) - N} \right\rceil. \quad (4-14)$$

Верхняя оценка. Простейшая верхняя оценка имеет следующий вид:

$$L^{верх} = \left\lceil \frac{\left\lceil \frac{2(h-1)}{K^{\min}} \right\rceil}{N} \right\rceil = \left\lceil \frac{2(h-1)}{NK^{\min}} \right\rceil, \quad (4-15)$$

где $K^{\min} = \min(K_N, \dots, K_1)$.

В работе [4] найдены условия, при выполнении которых эта оценка может быть снижена:

1. Если выполняется неравенство

$$K^{\min} \geq \frac{K^{\max}}{2} + 1, \quad (4-16)$$

то справедливо соотношение

$$L^{верх} \geq \left\lceil \frac{2(h-1)}{NK^{\max}} \right\rceil, \quad (4-17)$$

где $K^{\max} = \max(K_N, \dots, K_1)$.

Пример. Для набора модулей $K_1 = 4; K_2 = 3; K_3 = 3$ выполняется неравенство (4-16) и поэтому справедливо соотношение

$$L^{верх} = \left\lceil \frac{2(h-1)}{3 \times 4} \right\rceil.$$

Для набора модулей $K_1 = 4; K_2 = 3; K_3 = 2$ неравенство (4-16) не выполняется, и поэтому соотношением (4-17) в данном случае воспользоваться нельзя.

2. Если выполняются неравенства

$$\left. \begin{aligned} K^{\min} &\geq \left\lceil \frac{K^{\max}}{2} \right\rceil; \\ \frac{\sum_{i=1}^N K_i}{N} &\geq \frac{K^{\max}}{2} + 1, \end{aligned} \right\} (4-18)$$

то справедливо соотношение

$$L^{верх} = \left\lceil \frac{2(h-1)}{NK^{\max}} \right\rceil. \quad (4-19)$$

Пример. Для микросборки, содержащей модули $K_1 = 4; K_2 = 3; K_3 = 2$, выполняются неравенства (4-18), и поэтому

$$L^{верх} = \left\lceil \frac{2(h-1)}{3 \times 4} \right\rceil = \left\lceil \frac{h-1}{6} \right\rceil.$$

Интересно отметить, что для этой микросборки

$$K = \left(\sum_{i=1}^N K_i \right) - (N-1) = 7$$

и поэтому

$$L^{ниж} = \left\lceil \frac{h-1}{7-1} \right\rceil = \left\lceil \frac{h-1}{6} \right\rceil.$$

Таким образом,

$$L = \left\lceil \frac{h-1}{6} \right\rceil.$$

3. Если неравенства по пп. 1 и 2 не выполняются, то верхняя оценка может быть получена следующим образом: для некоторого подмножества N' модулей микросборки, включающего в себя модуль с K^{\min} , доказываем универсальность этого подмножества в классе формул, после чего оно рассматривается как новый модуль. Таким образом, формируется новый набор модулей, содержащий $N - N'$ исходных модулей и условно сформированный модуль. Выбор указанного подмножества производится так, чтобы условия, сформулированные в пп. 1 и 2, выполнялись при максимально,

возможном значении произведения $(N - N' - 1)(K^{\max})'$, где $(K^{\max})'$ - максимальное значение K для модулей нового набора.

Если после первого шага формирования набора условия, указанные в пп. 1 и 2, не выполняются, то процедура по пункту 3 может быть повторена многократно, до тех пор пока эти условия не будут выполнены. При этом полученные значения верхней оценки не должны быть выше значения, вычисленного на основе соотношения (4-15).

Пример. Пусть задана микросборка, содержащая модули $K_1 = 4; K_2 = 2; K_3 = 2$. Определить верхнюю оценку сложности реализации схемы в базисе этой микросборки.

Для этого набора модулей условия, указанные в пп. 1 и 2, не выполняются. Поэтому воспользуемся процедурой, изложенной в п. 3.

Известно, что модули $K_2 = 2$ и $K_3 = 2$ реализуют любую формулу из трех букв, и поэтому можно считать, что имеется модуль, универсальный в классе формул из трех букв, $K_4 = 3$. Для нового набора $K_1 = 4; K_4 = 3$. Так как для этого набора выполняются условия п. 1, то поэтому

$$L^{\text{верх}} = \left\lceil \frac{2(h-1)}{2 \times 4} \right\rceil.$$

Необходимо отметить, что модули $K_1 = 4$ и $K_2 = 2$ позволяют реализовать любую формулу из пяти букв. Поэтому новый набор может быть сформирован также из модулей $K_5 = 5$ и $K_3 = 2$. Однако при этом условия пп. 1 и 2 все еще не выполняются и поэтому повторим процедуру по п. 3. Так как модули с $K_5 = 5$ и $K_3 = 2$ реализуют любую формулу из шести букв, то $K_6 = 6$ и

$$L^{\text{верх}} = \left\lceil \frac{2(h-1)}{6} \right\rceil.$$

Таким образом, для заданной микросборки должно использоваться предыдущее выражение, имеющее большее значение знаменателя.

Оценки сложности реализации булевой функции, заданной в табличной форме. Пусть задана булева функция n переменных в виде неполностью определенной таблицы состояний, число единиц в столбце значений которой равно q_1 , а число нулей q_0 . При этом $q = q_1 + q_0 \leq 2^n$. Определим верхнюю оценку сложности реализации такой функции, выраженную в числе K -универсальных модулей. В работе [26] предложен метод построения по неполностью определенной таблице состояний скобочной формулы, число букв в которой удовлетворяет неравенству

$$h \leq 2q_1 + q_0 - 2. \quad (4-20)$$

Таблица 4-2

x_1	x_2	x_3	x_4	x_5	F
1	0	0	0	0	1
0	1	1	0	1	1
0	0	1	0	1	1
0	0	1	0	0	1
0	0	0	1	1	0
0	1	1	1	0	0
1	0	1	0	1	0
1	0	1	1	0	0

На основании этого результата может быть предложен простой признак для определения факта несущественной зависимости таблично заданной функции от некоторых своих переменных. Так как число букв в формуле не может быть меньше числа переменных, от которых она зависит, то, если для заданной таблицы выполняется неравенство $n > 2q_1 + q_0 - 2$, по крайней мере от $n - 2q_1 - q_0 + 2$ переменных функция зависит несущественно.

Искомая оценка сложности может быть получена в результате подстановки выражения, представленного в правой части неравенства (4-20), в правую часть соотношения

$$L \leq \left\lceil \frac{2(2q_1 + q_0 - 3)}{K} \right\rceil. \quad (4-21)$$

Пример. Реализовать функцию, заданную табл. 4-2 на 3-универсальных модулях.

Так как в данном случае $q_1 = q_0 = 4$, а $k = 3$, то $L \leq \left\lceil \frac{2(2 \times 4 + 4 - 3)}{3} \right\rceil = 6$.

На основании алгоритма, предложенного в работе [26], для заданной таблицы может быть построена скобочная формула вида $F = x_1 \bar{x}_3 \vee \bar{x}_1 x_3 \bar{x}_4$.

Эта формула может быть реализована схемой из двух 3-универсальных модулей, что соответствует приведенной выше оценке сложности. В работе [48] был получен результат, близкий к изложенному выше. При этом было показано, что

$$h \leq \frac{3}{2}q - 2 \text{ при } q \geq 2. \quad (4-22)$$

Подставляя выражение, приведенное в правой части неравенства, в правую часть соотношения (4-2), получим следующую оценку:

$$L \leq \left\lceil \frac{3(q-2)}{K} \right\rceil. \quad (4-23)$$

Объединив неравенства (4-21) и (4-23), получим верхнюю оценку сложности реализации функции, заданной в виде таблицы состояний:

$$L \leq \min \left\{ \left\lceil \frac{2(2q_1 + q_0 - 3)}{K} \right\rceil; \left\lceil \frac{3(q-2)}{K} \right\rceil \right\}.$$

Оценки сложности реализации систем булевых формул.

Выше были рассмотрены оценки сложности реализаций одной булевой формулы в базисе K -универсальных модулей. Приведем аналогичные оценки для системы N булевых формул y_1, y_2, \dots, y_N , число букв в правых частях которых соответственно равно h_1, h_2, \dots, h_N :

$$\left\lceil \frac{H-N}{K-1} \right\rceil \leq L \leq \left\lceil 2 \left(\frac{H-N}{K} \right) \right\rceil + N - 1, \quad (4-25)$$

где $H = \sum_{i=1}^N h_i$.

Выражение в правой части является искомой верхней оценкой в классе произвольных схем, в то время как выражение в левой части является нижней оценкой лишь в классе схем, в которых сумма числа выходов и внутренних разветвлений, не относящихся к входным переменным, равно N .

Отметим, что при $H = h_i, N = 1$ найденное выражение сводится к выражению (4-2), полученному для случая реализации одной формулы.

Пример. Реализовать формулу

$$y = (x_1 \vee x_2)x_3x_4 \vee [(x_1 \vee x_2)x_3 \vee x_5 \vee x_6]x_7x_8x_9x_{10}x_{11}x_{12}x_{13}$$

на модулях, универсальных в классе формул в базисе $\{\&, \vee, \bar{\quad}\}$ из трех букв.

В этом случае $H = 16, N = 1$, поэтому

$$8 = \left\lceil \frac{16-1}{3-1} \right\rceil \leq L \leq \left\lceil \frac{2(16-1)}{3} \right\rceil + 1 - 1 = 10.$$

Число модулей в схеме при ее реализации совпадает с нижней оценкой. Пример. Реализовать формулу, рассмотренную в предыдущем примере в виде системы формул

$$z = (x_1 \vee x_2)x_3;$$

При этом $H = 15, N = 2$, поэтому

$$7 = \left\lceil \frac{15-2}{3-1} \right\rceil \leq L \leq \left\lfloor \frac{2(15-1)}{3} \right\rfloor + 2 - 1 = 11.$$

Число модулей в схеме, реализующей заданную систему формул, совпадает с нижней оценкой, т. е. равно 7.

Пример. Реализовать систему формул

$$y_1 = (x_1 \vee x_2)x_3 \vee x_4 \vee x_5;$$

$$y_2 = (x_1 \vee x_2)x_3x_4$$

на модулях, универсальных в классе формул в базисе $\{\&, \vee, \bar{}\}$ из трех букв. Для этой системы $N = 2, H = 9$, поэтому

$$4 = \left\lceil \frac{9-2}{3-1} \right\rceil \leq L \leq \left\lfloor \frac{2(9-2)}{3} \right\rfloor + 2 - 1 = 6.$$

Число модулей в схеме, реализующей заданную систему формул, совпадает с нижней оценкой.

Пример. Реализовать систему функций, рассмотренную в предыдущем примере при ее представлении в виде:

$$z = (x_1 \vee x_2)x_3;$$

$$y = z \vee x_4 \vee x_5;$$

$$y_2 = zx_4$$

в базисе тех же модулей.

В этом случае $N = 3, H = 8$, поэтому

$$3 = \left\lceil \frac{8-3}{3-1} \right\rceil \leq L \leq \left\lfloor \frac{2(8-3)}{3} \right\rfloor + 3 - 1 = 6.$$

Число модулей в схеме в этом случае равно трем.

Минимизация числа K -универсальных модулей. В настоящее время имеется большое число работ, посвященных проблеме минимизации числа элементов в схемах [43]. Однако во многих из них вместо решения задачи минимизации числа элементов решается задача минимизации булевых формул, подлежащих реализации. При этом в большинстве работ рассматриваемая задача сводится к нахождению для заданной функции минимальной ДНФ или минимальной скобочной формулы, т. е. формул с минимальным числом букв. Указанная подмена в постановке задачи обосновывается тем, что уменьшение числа букв в булевой формуле обычно приводит к соответствующему уменьшению числа элементов. Однако взаимно-однозначное соответствие между сложностью булевой формулы в базисе $\{\&, \vee, \bar{}\}$, выраженной в числе букв, и сложностью схемы, выраженной в числе элементов имеет место лишь при построении параллельно-последовательных схем в базисе одноконтатных реле.

С усложнением элементного базиса указанное соответствие становится все менее очевидным, и можно привести примеры, когда при уменьшении числа букв в реализуемой формуле число элементов в соответствующей схеме увеличивается.

Соотношение (4-2) устанавливает связь между числом букв в реализуемой формуле, функциональными свойствами используемых модулей и необходимым числом этих модулей. Из рассмотрения этого соотношения следует, что при фиксированном значении K число модулей в схеме зависит от числа букв в реализуемой формуле. Поэтому, уменьшая число букв в формуле, обычно удается уменьшить число столь сложных элементов, какими являются настраиваемые логические модули. Таким образом, и в данном случае задача минимизации числа модулей может быть сведена к минимизации числа букв в реализуемой формуле.

Следовательно, важным достоинством предложенных модулей является то, что для их рационального использования не требуется разрабатывать специальный аппарат минимизации, как это имело место, например, при появлении мажоритарных элементов. При этом для модулей, универсальных в классе формул в базисе $\{\&, \vee, \bar{}\}$, может использоваться аппарат классической

минимизации (например, метод Квайна - Мак-Класки с последующим вынесением переменных за скобки), а для модулей, универсальных в классе формул в базисе $\{\&, \vee, \bar{}, \oplus\}$, или модулей, универсальных в классе всех булевых функций, - аппарат минимизации, разработанный в теории однородных структур [69].

Рассмотрим соотношение (4-2) более подробно. При этом можно отметить, что число модулей в схеме зависит не только от числа букв h в формуле, но и от структуры реализуемой формулы. Действительно, наличие интервала, задаваемого верхней и нижней оценками, определяется тем, что при $h = const$ существует $S(h)$ различных PN -типов формул. Поэтому, выполняя минимизацию только по критерию числа букв, можно для одном и той же булевой функции получить формулы различной структуры с одинаковым числом букв, требующих для своей реализации различного числа модулей. Более того, можно добиться сокращения числа модулей в схеме путем увеличения числа букв в формуле.

Пример. Реализовать функцию, заданную формулой из восьми букв,

$$y = x_1[x_2(x_3x_4 \vee x_5x_6) \vee x_7x_8]$$

на 3-универсальных модулях. В этом случае

$$4 \leq L(8;3) \leq 5,$$

а число модулей, требующихся для ее реализации, совпадает с верхней оценкой. Попробуем уменьшить сложность реализации путем увеличения числа букв в формуле. Для этого раскроем внутренние скобки в формуле, увеличив суммарное число букв до девяти:

$$y = x_1(x_2x_3x_4 \vee x_2x_5x_6 \vee x_7x_8).$$

При этом

$$4 \leq L(9;3) \leq 6,$$

а число модулей, требующихся для ее реализации, равно четырем.

Таким образом, можно утверждать, что если некоторая функция имеет представление в виде формулы, неповторной в некотором базисе, то реализация ее в неповторной форме необязательно приводит к минимальному числу модулей.

У читателя может сложиться представление, что в общем случае совсем неясно, как проводить минимизацию и стоит ли это делать. Это неправильно. Отметим прежде всего, что продемонстрированный «фокус» получается только при изменении структуры выражения, например при переходе к скобочным формам. Сокращение числа букв в ДНФ всегда будет способствовать уменьшению числа модулей. Осторожность следует проявлять только при вынесении за скобки: в итоге фрагменты могут стать такими мелкими, что возможности модуля будут использоваться неполностью.

При некотором опыте разработчик не станет «портить» фрагменты, однако для безопасности приведем условия, при которых сокращение числа букв в формуле с гарантией не приведет к возрастанию числа модулей в схеме.

Пусть булева функция, заданная формулой из h_1 букв, реализуется схемой из L_1 модулей. Выполним минимизацию этой формулы, и пусть число букв в формуле после минимизации равно h_2 , а необходимое число модулей, требующееся для ее реализации, L_2 .

Реализуем эти формулы в классе структур, в которых разветвления допустимы лишь на уровне входных переменных; тогда, если справедливо соотношение

$$h_2 \leq \left[1 + \frac{h_1 - 1}{2} \times \frac{K}{K - 1} \right], \quad (4-26)$$

число модулей в схеме не увеличивается.

Предположим, что $h_1 = 55, K = 7$. Тогда, если в результате минимизации число букв в формуле удастся сократить до

$$h_2 \leq \left[1 + \frac{55 - 1}{2} \times \frac{7}{7 - 1} \right] = 32,$$

число модулей в схеме, реализующей минимизированную формулу, не возрастет.

Если выполняется соотношение

$$h_2 \leq \left[1 + \frac{h_1 - 1}{2} \cdot \frac{K}{K - 1} \right] - \left] \frac{K}{2} \right[, \quad (4-27)$$

то число модулей в схеме уменьшается.

Пусть $h_1 = 73, K = 5$. Тогда, если в результате минимизации число букв в формуле удастся уменьшить до

$$h_2 \leq \left[1 + \frac{73 - 1}{2} \times \frac{5}{5 - 1} \right] - \left] \frac{5}{2} \right[= 43,$$

число модулей в схеме, реализующей минимизированную формулу, точно уменьшится.

Соотношения для системы булевых формул, аналогичные неравенствам (4-26) и (4-27), имеют вид

$$H_2 \leq \left[N_2 + \left(1 + \frac{H_1 - N_1}{K - 1} - N_2 \right) \cdot \frac{K}{2} \right]; \quad (4-29)$$

$$H_2 \leq \left[N_2 + \left(1 + \frac{H_1 - N_1}{K - 1} - N_2 \right) \cdot \frac{K}{2} \right] - \left] \frac{K}{2} \right[. \quad (4-29)$$

ГЛАВА ПЯТАЯ

ЦИФРОВЫЕ ИНТЕГРАЛЬНЫЕ МИКРОСХЕМЫ ШИРОКОГО ПРИМЕНЕНИЯ КАК НАСТРАИВАЕМЫЕ МОДУЛИ

5-1. ОЦЕНКА ЛОГИЧЕСКОЙ ЭФФЕКТИВНОСТИ МИКРОСХЕМ

Обоснованный выбор интегральных микросхем (ИМС) из числа существующих, а также теоретические основания для построения новых невозможны без разработки и использования критериев их логической эффективности.

ЦИФРОВЫЕ ИНТЕГРАЛЬНЫЕ МИКРОСХЕМЫ ШИРОКОГО ПРИМЕНЕНИЯ КАК НАСТРАИВАЕМЫЕ МОДУЛИ

5-1. ОЦЕНКА ЛОГИЧЕСКОЙ ЭФФЕКТИВНОСТИ МИКРОСХЕМ

Обоснованный выбор интегральных микросхем (ИМС) из числа существующих, а также теоретические основания для построения новых невозможны без разработки и использования критериев их логической эффективности.

Разрабатываемые критерии должны: а) иметь ясный физический смысл, т. е. быть связанными с той или иной интересующей нас характеристикой предстоящей схемной реализации (например, с числом корпусов ИМС); б) быть конструктивными - позволять целенаправленно влиять на характеристики схем, ими определяемые; в) учитывать свойства используемых логических элементов и реализуемых алгоритмов функционирования; г) учитывать особенности используемых методов синтеза, так как логический модуль является эффективным лишь тогда, когда имеется метод синтеза схем на таких модулях, позволяющий использовать все его логические возможности.

Традиционный способ оценки качества выбора логических элементов основан на подсчете их числа, потребного для реализации некоторого гипотетического устройства [53]. С другой стороны, в ряде работ [16, 34, 37] эффективность модулей оценивается числом подфункций, которые могут быть получены; при настройке. Однако большое число подфункций само по себе еще не свидетельствует о высокой эффективности модуля, так как доля числа подфункций, реализуемых модулем, по отношению к теоретически возможному числу функций от того же числа переменных может быть при этом чрезвычайно малой. Кроме того, введенные в этих работах коэффициенты обладают и тем недостатком, что они не входят в оценки числа модулей, требующихся для предстоящей реализации заданной формулы. Поэтому, например, при оценке логической эффективности некоторого модуля с числом входов M - числом порождаемых им подфункций - остается неясным, сколько же таких модулей понадобится для реализации конкретной формулы из h букв в случае, если $h > M$.

Кроме того, неясно и другое: если некоторый другой модуль порождает меньшее число подфункций, то потребуются ли для реализации заданной формулы таких модулей больше, чем модулей, более эффективных по этому критерию?

В данном параграфе предлагается критерий логической эффективности, позволяющий устранить указанные недостатки и удовлетворяющий требованиям, приведенным выше.

Для класса K -универсальных модулей, рассмотренных в гл. 3,1 такой критерий может быть найден, если их логическую эффективность оценивать значением коэффициента K - числом букв, для которого рассматриваемый модуль универсален в классе формул в некотором базисе. Этот коэффициент входит в знаменатель оценок сложности предстоящей реализации:

$$\left\lceil \frac{h-1}{K-1} \right\rceil \leq L \leq \left\lceil \frac{2(h-1)}{K} \right\rceil, \quad (5-1)$$

и поэтому можно утверждать, что он достаточно объективно характеризует логические возможности модуля, так как чем выше его значение (чем от большего числа букв модуль универсален), тем меньшее число этих модулей требуется для реализации произвольной формулы из h букв.

Кроме того, введенный показатель обладает и тем свойством, что если имеется два модуля с коэффициентами логической эффективности K_1 и K_2 , причем $K_1 > K_2$, то не может быть найдено ни одной формулы ни при каком числе букв в ней, для реализации которой потребуется модулей первого типа больше, чем модулей второго типа.

Введем коэффициент логической эффективности многофункционального модуля таким образом, чтобы с его помощью логическая эффективность универсального модуля могла быть определена как частный случай. Предполагая, что любая формула в массиве реализуемых формул от фиксированного числа букв равновероятна, определим коэффициент логической эффективности многофункционального модуля

$$K_s = \sum_{i=1}^v \frac{\Phi(i)}{S(i)}, \quad (5-2)$$

где $\Phi(i)$ - число PN -типов неповторных формул из i букв, реализуемых модулем при настройке; v - максимальное число букв, для которого модуль при настройке реализует неповторные формулы; $S(i)$ - число PN -типов неповторных формул из i букв. Определим значение этого коэффициента для модуля, структура которого описывается формулой $y = x_1x_2 \vee x_3x_4$. Этот модуль при настройке реализует: одного представителя PN -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из одной буквы (1), двух представителей типов формул из двух букв (1 + 1; 2), двух из четырех представителей типов формул из трех букв (2+1; 1 + 1) и одного из возможных десяти представителей типов формул из четырех букв (2 + 2). Поэтому для этого модуля

$$K_s = \frac{1}{1} + \frac{2}{2} + \frac{2}{4} + \frac{1}{10} = 2,6.$$

Введенный коэффициент логической эффективности может интерпретироваться как эквивалентное число информационных входов. Поэтому, несмотря на то что рассмотренный модуль имеет четыре входа, при его применении для достаточно большого массива формул у него удастся использовать в среднем лишь 2,6 входов, а остальные входы используются для настройки.

Необходимо отметить, что так как K - универсальный модуль реализует при настройке все неповторные формулы из 1, 2, ..., K букв, то для него $K_s = K$, и поэтому введенный коэффициент может использоваться в усредненной оценке сложности

предстоящей реализации в базисе многофункциональных модулей, которая при этом имеет вид

$$\left\lceil \frac{h-1}{K_s-1} \right\rceil \leq L' \leq \left\lceil \frac{2(h-1)}{K_s} \right\rceil. \quad (5-3)$$

Таким образом, предложенный коэффициент логической эффективности многофункциональных модулей обладает тем свойством, что входит в оценку сложности предстоящей реализации булевых формул в базисе таких модулей. Это позволяет установить связь между логической эффективностью модулей и их числом, необходимым для реализации заданной булевой формулы, что не удавалось сделать при использовании других показателей оценки логической эффективности.

Предложенный коэффициент может быть использован для оценки функциональных возможностей практически любого комбинационного логического элемента или серии комбинационных логических элементов, рассматриваемых при этом как многофункциональный логический модуль. Действительно, любая комбинационная схема с v входами может рассматриваться как многофункциональный модуль, так как при его настройке (подаче констант и отождествлении входов) она способна реализовать различные булевы формулы от меньшего числа букв. Аналогично, серия логических элементов, реализующих соответственно различные формулы f_1, f_2, \dots, f_a , может условно рассматриваться как один многофункциональный модуль, имеющий $\beta > \alpha$ настроек.

Введенный коэффициент логической эффективности, ориентированный на класс неповторных формул, может быть использован для широкого класса ИМС, выпускаемых промышленностью, структура которых описывается формулами, неповторными в базисе $\{\&, \vee, \bar{\quad}\}$. Этот же коэффициент может применяться также и для модулей, структура которых описывается повторными формулами, если эти модули в результате настройки реализуют неповторные формулы. В качестве примера таких элементов могут быть приведены многофункциональные логические модули, входящие в состав серий 108 и 511 [59].

Рассмотрим в качестве примера два модуля, структура первого из которых описывается формулой

$$y_1 = x_1x_2 \vee x_3x_4 \vee x_5x_6 \vee x_7x_8x_9,$$

а второго -

$$y_2 = x_1x_2 \vee x_3x_4 \vee x_5x_6x_7 \vee x_8x_9x_{10}x_{11}.$$

Эти модули (К1ЛР333 и К1ЛР342) являются наиболее сложными из комбинационных схем, входящих соответственно в состав серий 133 и 134. Оценим логические возможности этих ИМС. Для

этого для каждой из них определим представителей всех типов неповторных формул в базе $\{\&, \vee, \bar{}\}$, реализуемых при настройке.

В табл. 5-1 приведены значения $\Phi(i)$ для рассматриваемых ИМС, а также значения $S(i)$.

Определим значения $K_{\text{с}}$ для рассматриваемых ИМС:

$$K_{\text{с}1} = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{8}{10} + \frac{10}{24} + \frac{12}{66} + \frac{9}{180} + \frac{4}{522} + \frac{1}{1532} = 4,46;$$

$$K_{\text{с}2} = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{10}{10} + \frac{17}{24} + \frac{26}{66} + \frac{32}{180} + \frac{30}{522} + \frac{18}{1532} + \frac{7}{4624} + \frac{1}{14136} = 5,35.$$

Оценим теперь функциональные возможности ИМС серии 108 [51]. Для реализации комбинационных схем в состав этой серии введен многофункциональный логический модуль К1ЖЛ081, рассмотренный в § 3-4. Структура

Таблица 5-1

Функциональные возможности микросхем К1ЛР333 и К1ЛР342

i	1	2	3	4	5	6	7	8	9	10	11
$\Phi_1(i)$	1	2	4	8	10	12	9	4	1	-	-
$\Phi_2(i)$	1	2	4	10	17	26	32	30	18	7	1
$S(i)$	1	2	4	10	24	66	180	522	1532	4624	14136

Этого модуля, имеющего два выхода – прямой и инверсный, описывается следующими повторными формулами:

$$F_1(x_1, \dots, x_{10}) = x_1 x_2 x_3 x_4 x_5 x_6 x_7 \bar{x}_8 \bar{x}_9 \bar{x}_{10} \vee (x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 x_8) \bar{x}_9 \bar{x}_{10} \vee (x_1 x_2 x_3 x_4 \vee x_5 x_6 x_7 x_8) \bar{x}_9 \bar{x}_{10} \vee (x_1 x_2 x_3 \vee x_4 x_5 x_6 \vee x_7 \vee x_8 x_9 x_{10});$$

$$F_2(x_1, \dots, x_{10}) = \bar{F}_1(x_1, \dots, x_{10}).$$

В состав этой серии входит также кворум-элемент «3 из 4» (К.1ЛР081), описываемый формулой

$$F(x_1, \dots, x_8) = (x_1 \vee x_2) x_3 x_4 \vee (x_5 \vee x_6) x_7 x_8.$$

Микросхема К1ЖЛ081 реализует при настройке большое число различных функций, однако, поскольку общее число произвольных булевых функций

растет, как 2^{2^n} , то вероятность использования этой ИМС в классе всех булевых функций от сравнительно большого числа входов практически равна нулю. Поэтому рассмотрим функциональные возможности этой микросхемы в классе неповторных формул в базе $\{\&, \vee, \bar{}\}$. Одновременно оценим также логическую эффективность ее при добавлении кворум-элемента.

В работе [5] содержится таблица, в которой приведены все неповторные формулы, реализуемые этими ИМС при настройке. Значения $\Phi_j(i)$ для микросхемы К1ЖЛ081 [$\Phi_3(i)$] и для совместного использования этой микросхемы и микросхемы К1ЛР081 [$\Phi_4(i)$] даны в табл. 5-2.

При этом

$$K_{\text{с}3} = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{10}{10} + \frac{24}{24} + \frac{58}{66} + \frac{50}{180} + \frac{26}{522} = 6,21;$$

$$K_{\text{с}4} = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{10}{10} + \frac{24}{24} + \frac{61}{66} + \frac{52}{180} + \frac{27}{522} = 6,265.$$

Из приведенных результатов следует, что рассматриваемый модуль универсален в классе неповторных формул из пяти и менее букв, так как он способен реализовать любую формулу в базе $\{\&, \vee, \bar{}\}$ из этого числа букв при равнодоступности прямого и инверсного выходов ИИ. Поэтому реализация любой формулы из пяти и менее букв, заданной в этом базисе, на одном таком модуле гарантирована. Таким образом, в этом случае

$$L \leq \left\lceil \frac{2(h-1)}{5} \right\rceil.$$

Из табл. 5-2 следует также, что расширение базиса серии за счет использования кворум-элемента как многофункционального модуля практически

Таблица 5-2

Функциональные возможности ИМС серии 108

i	1	2	3	4	5	6	7	8	9	10
$\Phi_3(i)$	1	2	4	10	24	58	50	26	-	-
$\Phi_4(i)$	1	2	4	10	24	61	52	27	-	-

не увеличивает логическую эффективность серии и не обеспечивает ее универсальности при реализации формул в базисе $\{\&, \vee, \bar{}\}$ из шести букв.

В заключение этого параграфа рассмотрим вопрос об оценке логических возможностей модулей в классе неповторных ДНФ. При этом необходимо отметить что модуль, универсальный в классе неповторных формул в базисе $\{\&, \vee, \bar{}\}$ из K_1 букв, может быть одновременно универсальным в классе неповторных ДНФ из K_2 букв, где $K_2 > K_1$, что в ряде случаев может позволить снизить верхнюю оценку сложности.

Так, исследование функциональных возможностей микросхемы ЮЖЛ081 в классе неповторных ДНФ показало, что она универсальна в этом классе

Таблица 5-3

Функциональные возможности микросхемы К1ЖЛ081 в классе неповторных ДНФ

i	1	2	3	4	5	6	7	8
$\Phi_5(i)$	1	2	3	5	7	11	9	8
$S_1(i)$	1	2	3	5	7	11	15	22

из шести и менее букв при равнодоступности прямых и инверсных выходов ИИ (табл. 5-3). В таблице приведены также теоретические значения числа PN -типов неповторных ДНФ. При этом

$$K'_9 = \frac{1}{1} + \frac{2}{2} + \frac{3}{3} + \frac{5}{5} + \frac{7}{7} + \frac{11}{11} + \frac{9}{15} + \frac{8}{22} = 6,964.$$

Связь между функциональными возможностями модулей, универсальных в классе ДНФ из K' букв, и числом этих модулей, требующимся для реализации произвольной формулы в базисе $\{\&, \vee, \bar{}\}$ из h букв, число скобок в которой равно c , устанавливается, как было показано в гл. 4, соотношением

$$L \leq \left\lceil \frac{h-1}{\left\lceil \frac{K'}{2} + 1 \right\rceil} \right\rceil + \frac{c}{2}. \quad (5-3)$$

Для рассматриваемой ИМС это соотношение имеет вид

$$L \leq \left\lceil \frac{h-1}{4} \right\rceil + \frac{c}{2}.$$

Таким образом,

$$L \leq \min \left\{ \left\lceil \frac{2(h-1)}{5} \right\rceil; \left\lceil \frac{h-1}{4} \right\rceil + \frac{c}{2} \right\}.$$

5-2. ПОСТРОЕНИЕ ЛОГИЧЕСКИХ СХЕМ НА МИКРОСХЕМАХ

Применение предложенных в гл. 4 методов реализации логических устройств на НЛМ из ФЭ не предполагает обязательного использования специально сформированных модулей. Действительно, для

использования этих методов требуется иметь элементы, реализующие необходимую номенклатуру неповторных формул из определенного числа букв. Эти формулы могут реализовываться модулем одного типа, но возможен вариант использования набора стандартных реализаций этих формул. Каждая стандартная реализация может формироваться из любых серийно выпускаемых функциональных элементов, и в том числе из ИМС.

Обладая набором стандартных реализаций всех представителей типов неповторных формул из K и менее букв, нетрудно построить схему, соответствующую формуле длиной до K букв, так как для этого достаточно выбрать требуемую схему из упомянутого набора (каталога). Если реализуемая формула содержит более K букв, то она может быть построена на базе метода, изложенного в гл. 4, в виде схемы из стандартных реализаций. Таким образом, для рационального использования серий ИМС, выпускаемых промышленностью, необходимо предварительно исследовать их функциональные возможности с целью построения таблиц стандартных оптимальных реализаций представителей всех PN -типов формул, неповторных в базисе $\{\&, \vee, \bar{}\}$, для возможно большего числа букв K^{max} .

В процессе построения указанных таблиц путем перебора выполняется условное расширение базиса серии за счет построения из вентилях, входящих в микросхемы серии, всех возможных одновыходных схем без обратных связей. Максимальная сложность каждой схемы не превосходит наперед заданного числа корпусов, определяемого допустимым объемом выполняемого перебора.

Необходимо отметить, что указанная процедура исследования функциональных возможностей при фиксированной номенклатуре серии выполняется лишь однажды, а затем осуществляется блочный синтез с использованием найденных стандартных оптимальных реализаций.

Наличие таблиц оптимальных реализаций представителей всех PN -типов неповторных формул в указанном базисе для K^{max} и менее букв позволяет осуществлять реализацию формул этого класса путем выбора из таблицы соответствующей минимальной схемы, и, следовательно, чем больше значение K^{max} , для которого построены таблицы, тем реже приходится выполнять собственно процедуру синтеза. Для формул, число букв в которых больше K^{max} , осуществляется синтез с использованием построенных таблиц.

Изложив общий подход к рациональному использованию серий рассмотрим этот вопрос применительно к микросхемам серии 133, наиболее широко применяемым в настоящее время при построении управляющих логических устройств на микроэлектронной элементной базе.

При этом воспользуемся ИМС следующих шести типов: К.1ЛБ333, К.1ЛБ334, К1ЛБ331, К1ЛБ332, К1ЛР331, К1ЛР333 [51]. Первые четыре ИМС содержат вентили «И - НЕ» на два, три, четыре и восемь входов, а две остальные имеют более высокий уровень схемной интеграции: первая из них содержит два вентиля, структура которых описывается формулой $y = x_1x_2 \vee x_3x_4$, а вторая - один вентиль

$$y = \overline{x_1x_2 \vee x_3x_4 \vee x_5x_6 \vee x_7x_8x_9}.$$

Излагаемый подход базируется на следующих основных положениях:

- а) рассматриваемые элементы имеют структуру, описываемую неповторными формулами в базисе $\{\&, \vee, \bar{}\}$;
- б) рациональное использование элементов серии требует предварительного исследования всех функциональных возможностей в классе неповторных формул в базисе $\{\&, \vee, \bar{}\}$ каждого из элементов серии и схем из них, рассматриваемых как новые элементы серии;
- в) любой логический элемент серии, имеющий более одного входа, или схема из логических элементов рассматривается как многофункциональный модуль, настраиваемый на реализацию неповторных формул в базисе $\{\&, \vee, \bar{}\}$.
- г) вся совокупность элементов, входящих в серию, и схемы из них условно рассматриваются как один многофункциональный модуль, настраиваемый на реализацию неповторных формул;
- д) для каждой неповторной формулы, реализуемой модулем в результате настройки, приводится в специальной таблице простейшая ее реализация в базисе рассматриваемой серии;
- е) настройка этого условного модуля осуществляется сочетанием следующих способов: выбором требуемой схемы или элемента серии; приравнением части входных сигналов нулю или единице; объединением входов.

На основе изложенного подхода было найдено путем перебора более 90 объединенных в каталог схем из вентиляей, входящих в эти ИМС. Конструктивная сложность этих схем не превышает $1/4$ корпуса.

Для каждой из схем были найдены все неповторные формулы в базисе $\{\&, \vee, \bar{}\}$, реализуемые при настройке, и для каждой такой формулы были выбраны простейшие реализации, в результате чего была сформирована таблица оптимальных реализаций.

Эта таблица позволяет найти для каждого PN -типа неповторных формул в базисе $\{\&, \vee, \bar{}\}$ из шести и менее букв, а также для некоторых типов формул для большего числа букв по крайней мере по одному представителю, реализуемому оптимально.

Использование построенных таблиц позволяет без синтеза находить минимальные реализации произвольных формул в базисе $\{\&, \vee, \bar{}\}$ из шести и менее букв при условии равнодоступности прямых и инверсных выходов III , а также из трех и менее букв при условии неравнодоступности выходов III .

Эти таблицы являются также основой для определения оценок сложности предстоящих реализаций в базисе ИМС примеряемой серии.

Число корпусов, требующихся для реализации формулы, заданной в базисе $\{\&, \vee, \bar{}\}$ из h букв, при использовании PN -классификации удовлетворяет неравенству

$$L \leq \min_K \left\{ a_K \left[\frac{2(h-1)}{k} \right] \right\}, \quad (5-4)$$

где a_K - часть корпуса, требующегося для реализации представителя наиболее сложного PN -типа неповторных формул в указанном базисе из K букв.

Зафиксируем для построенной таблицы значение $a_K \max$; тогда ввиду того что увеличение числа букв в формуле на одну не может привести к увеличению $a_K \max$ более чем на a_2 корпусов, для формулы из h букв будет выполняться соотношение

$$L \leq]a_K \max + ia_2[, \quad (5-5)$$

где

$$i = h - K^{\max}.$$

Таким образом, объединяя найденные соотношения, получим

$$L \leq \min \left(\min_K \left\{ a_K \left[\frac{2(h-1)}{k} \right] \right\}; a_K \max + ia_2 \right). \quad (5-6)$$

Выше отмечалось, что использование PN -классификации возможно лишь в случаях, когда прямые и инверсные выходы III равнодоступны, или при применении инверторов. При неравнодоступности выходов и отсутствии инверторов должна использоваться P -классификация. При этом в приведенные выше соотношения вместо коэффициента a_K должен подставляться коэффициент b_K - часть корпуса, требующегося для реализации представителя наиболее сложного P -типа неповторных формул в указанном базисе из K букв.

Найденные оценки могут быть использованы также в качестве верхних оценок величины

$$L' = \sum_{i=1}^L \frac{t_{i3}}{t_i} \quad (5-7)$$

при замене коэффициента $a_K (b_K)$ в соотношении (5-6) на коэффициент $a'_K (b'_K)$. При этом: t_{i3} - число задействованных вентиляей в i -й микросхеме; t_i - общее число вентиляей в i -й микросхеме; $a'_K (b'_K)$ - значение показателя L' для представителя наиболее сложного PN -типа (P -типа) формул, неповторных в базисе $\{\&, \vee, \bar{}\}$ из K букв.

Значения введенных коэффициентов для микросхем серии 133, полученные на основе анализа таблицы оптимальных реализаций, приведены в табл. 5-4.

Более подробное исследование таблиц оптимальных реализаций позволило определить значения коэффициентов K и K' для ИМС рассматриваемой серии при сложности реализации, не превышающей одного корпуса:

$$K = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{10}{10} + \frac{22}{24} + \frac{24}{66} + \frac{14}{180} + \frac{5}{522} + \frac{1}{1532} = 5,37;$$

$$K' = \frac{1}{1} + \frac{2}{2} + \frac{4}{4} + \frac{10}{10} + \frac{24}{24} + \frac{35}{66} + \frac{16}{180} + \frac{5}{522} + \frac{1}{1532} = 5,63.$$

Перейдем к изложению процедуры реализации нормальной булевой формулы, заданной в базисе $\{\&, \vee, \bar{}\}$, на основе таблицы оптимальных реализаций. Эта процедура не гарантирует

Таблица 5-4

Значения коэффициентов в оценках сложности предстоящих реализаций для микросхем серии 133

K	2	3	4	5	6
b_K	$\frac{3}{4}$	-	-	-	-
a_K	$\frac{1}{2}$	$\frac{2}{3}$	1	-	-
b'_K	$\frac{3}{4}$	1	-	-	-
a'_K	$\frac{1}{2}$	$\frac{7}{12}$	$\frac{5}{6}$	1	$1\frac{1}{4}$

нахождения минимальных решений при $K > 6$, но позволяет строить схемы, сложность которых не превосходит верхних оценок, указанных выше.

Процедура реализации.

1. Определяются оценки сложности предстоящей реализации формулы.
2. Заданная формула представляется в виде арифметического полинома.
3. Для найденного арифметического полинома строится структурная схема, каждый элемент которой описывается арифметическим полиномом, совпадающим с полиномом наибольшей длины из имеющихся в таблице. При этом, если при построении структурной схемы используются лишь арифметические полиномы длиной K^{max} и менее, то сходимость процесса реализации гарантирована.
4. Структурная схема преобразуется от выхода к входу. При этом арифметический полином выходного элемента не изменяется и для него выбирается формула, имеющая простейшую реализацию из всех формул, приведенных в таблице для этого полинома. Выбор булевых формул для элементов последующих слоев осуществляется по правилу: если от выхода рассматриваемого элемента до выхода схемы (после выбора для всех элементов, соединяющих выход этого элемента с выходом схемы, формул, имеющих простейшие реализации) сигнал «проходит» четное число инверсий, то его арифметический полином не изменяется, и для него выбирается формула, имеющая простейшую реализацию из всех формул, приведенных в таблице для этого полинома; если сигнал «проходит» нечетное число инверсий, то арифметический полином заменяется на «дополнительный» и для нового полинома выбирается формула, имеющая простейшую реализацию в указанном выше смысле.

В преобразованной структурной схеме для каждого элемента оказывается арифметический полином, кружками отмечаются инверсные входы и приводится номер структуры, реализующей этот полином.

5. Каждый элемент преобразованной структурной схемы заменяется оптимальной его реализацией в соответствии с номером в таблице, указанным при этом элементе. В результате формируется схема, состоящая из вентилях используемой серии.
6. В построенной схеме устраняется избыточность, существующая в случае, если два инвертора оказываются включенными последовательно.
7. Определяется формула, реализуемая с помощью предлагаемой процедуры. Для этого от входа к выходу записывается формула, соответствующая построенной схеме. Найденная таким образом формула не принадлежит к классу нормальных, и поэтому с целью устранения инверсий над группами переменных в ней подсчитывается их число над каждой буквой и каждым знаком операции; при нечетном их числе соответствующая буква в нормальной формуле используется с инверсией, а знак операции изменяется на дополнительный.
8. Если полученная формула совпадает с заданной, то процедура логического синтеза завершается. Если полученная формула отличается от заданной расстановкой инверсий над одиночными символами

переменных, то при равнодоступности прямых и инверсных выходов *III* процедура логического синтеза также заканчивается, а в случае неравнодоступности выходов *III* на входах схемы, для которых имеет место указанное различие, устанавливаются инверторы, после чего процедура логического синтеза завершается.

9. Вентили, входящие в построенную схему, распределяются по корпусам ИМС и объединяются между собой в соответствии с найденной схемой.

10. Определяются характеристики построенной схемы L' и L .

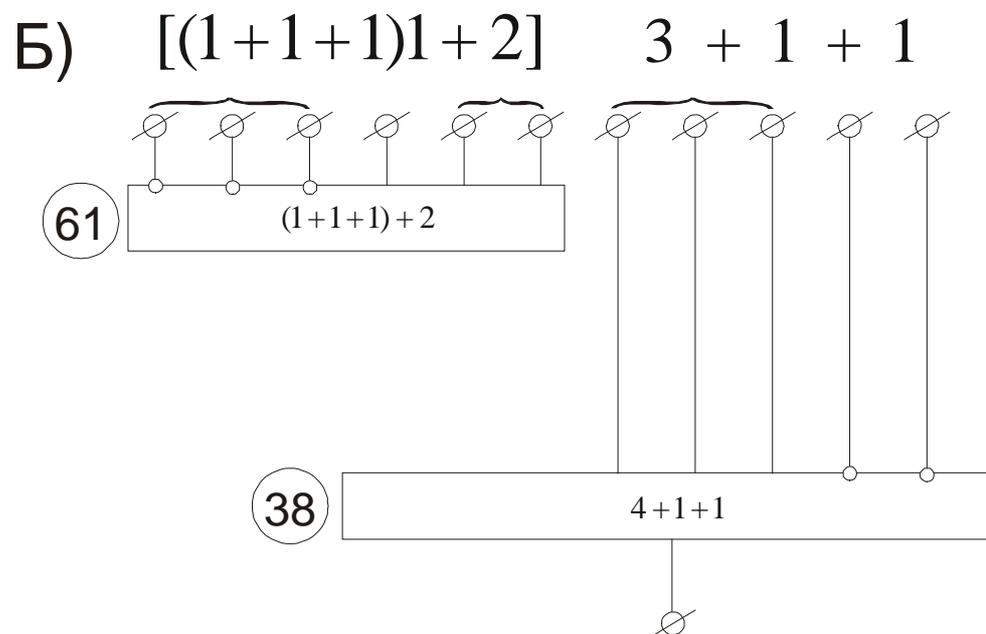
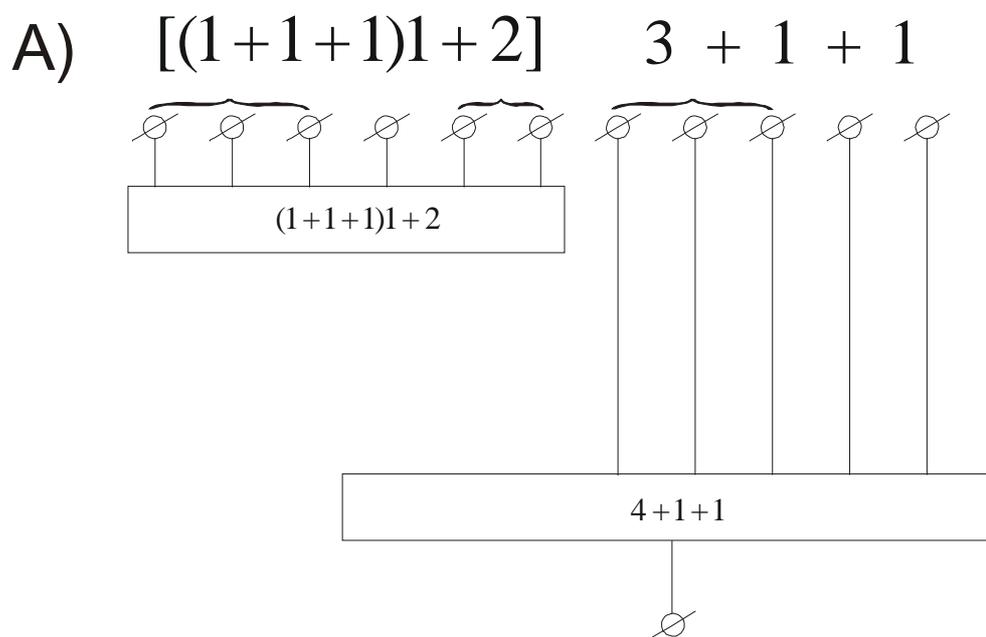
11. Изложенная процедура ориентирована на минимизацию показателя L' . В ряде случаев удается после построения схемы уменьшить число использованных микросхем L за счет увеличения значения показателя L' . Это возможно в тех случаях, когда в построенной схеме используются вентили «И-НЕ» и эти вентили, имеющие малое число входов, удастся распределить по незадействованным вентилям «И - НЕ», имеющим большее число входов, так что число использованных микросхем уменьшается.

12. Определяются характеристики нового размещения схемы по выбранным корпусам ИМС.

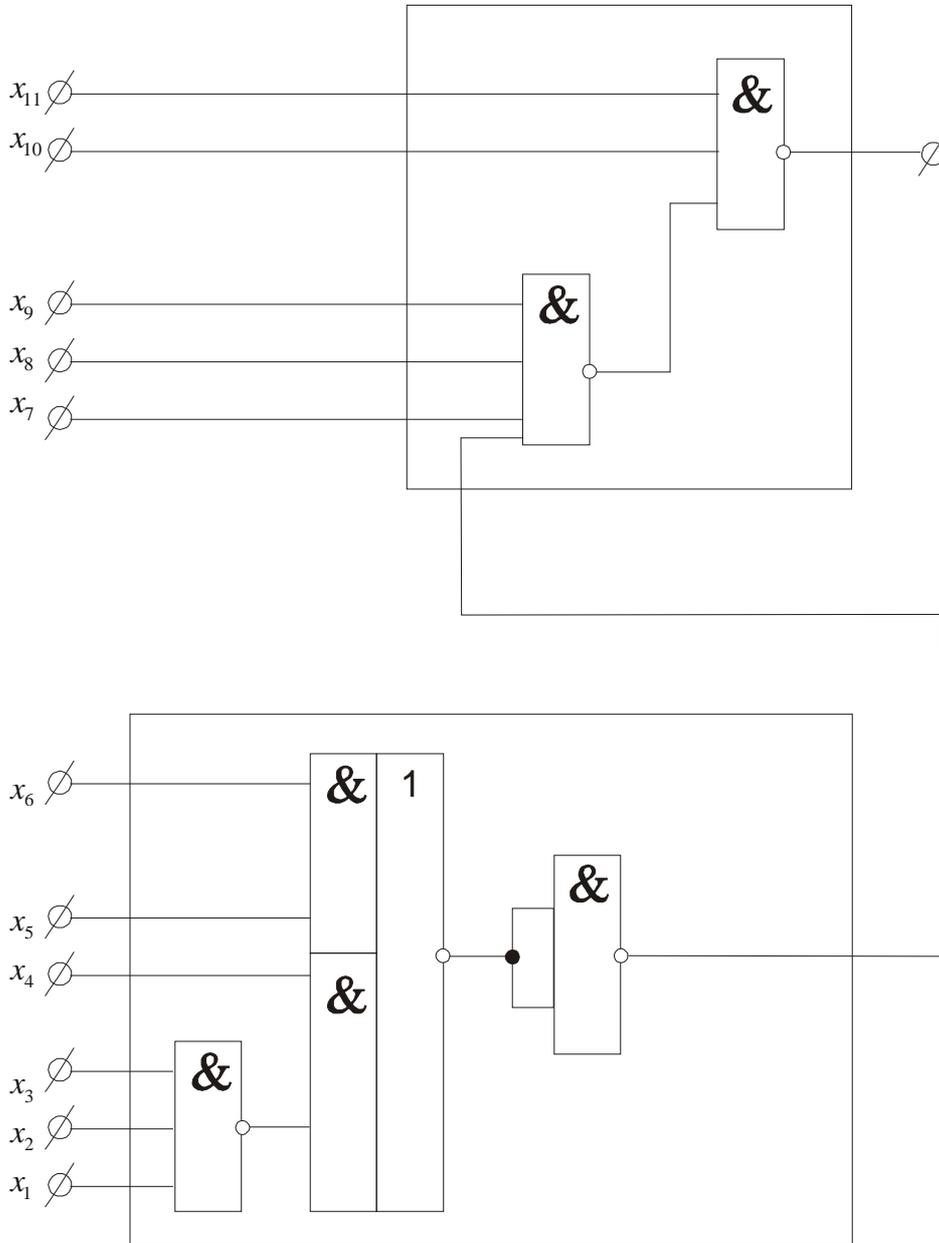
Пример. Реализовать формулу

$$y = (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)x_4 \vee x_5x_6]x_7x_8x_9 \vee \bar{x}_{10} \vee \bar{x}_{11}$$

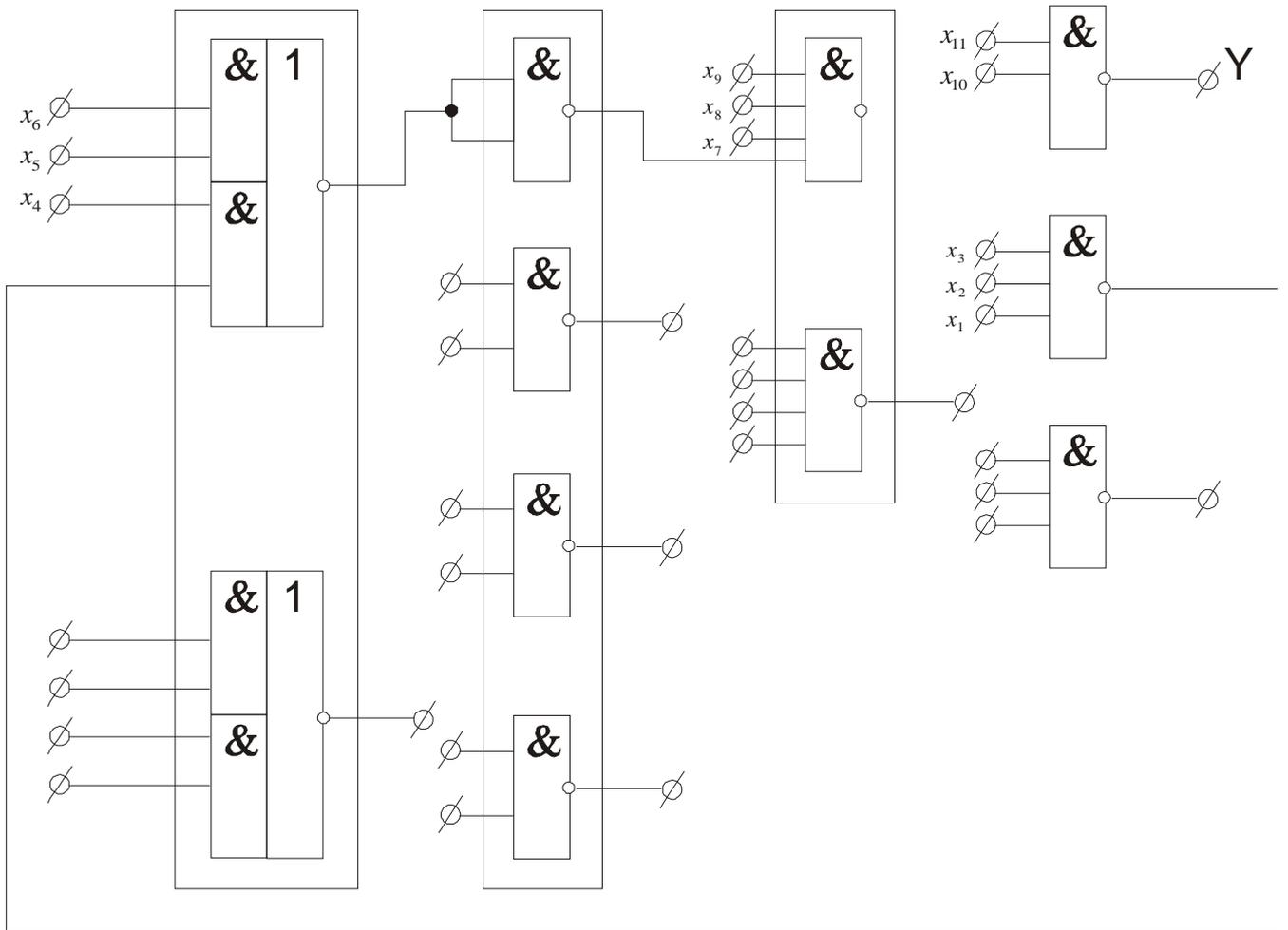
на ИМС серии 133.



B)



Г)



Д)

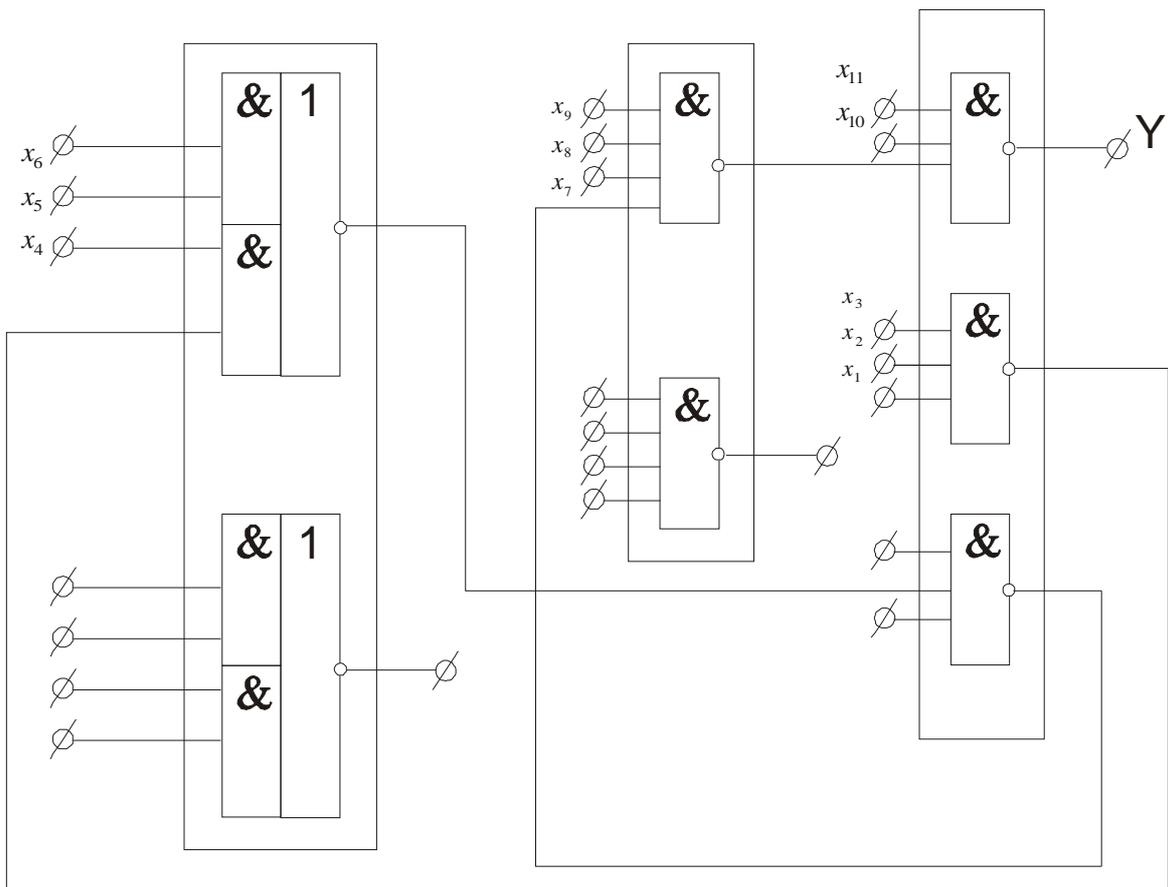


Рис. 5-1. Реализация булевой формулы $y = [(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)x_4 \vee x_5x_6]x_7x_8x_9 \vee \bar{x}_{10} \vee \bar{x}_{11}$ на микросхемах серии 133: *a* - структурная схема; *б* - скорректированная структурная схема; *в* - схема из вентилях микросхем серии . *г* – распределение вентилях построенной схемы по корпусам микросхем; *д* - перераспределение вентилях построенной схемы по корпусам микросхем

1. $L' \leq 3\frac{3}{4}; L \leq 5.$

2. $[(1+1+1)1+2]3+1+1.$

3. Рис. 5-1,а.

4. Рис. 5-1,б.

5. Рис. 5-1,в.

6. Избыточность отсутствуют.

7. $x_1x_2x_3x_4 \vee x_5x_6x_7x_8x_9x_{10}x_{11} = [(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)x_4 \vee x_5x_6]x_7x_8x_9 \vee \bar{x}_{10} \vee \bar{x}_{11}.$

8. Полученная формула совпадает с заданной. Логический синтез завершен.

9. Рис. 5-1,г.

10. Характеристики построенной схемы:

$$L' = \frac{2}{3} + \frac{1}{2} + \frac{1}{2} + \frac{1}{4} = 1\frac{1}{12}; L = 4.$$

11. Рис. 5-1,д.

12. $L' = 1 + \frac{1}{2} + \frac{1}{2} = 2; L = 3.$

Примечание. При реализации заданной формулы на двухвходовых вентилях «И - НЕ» построенная схема характеризуется следующими показателями: $L' = 3,5; L = 4[2].$

5-3. УНИФИКАЦИЯ БЛОКОВ ЛОГИК ПОСТРОЕННЫХ ИЗ МИКРОСХЕМ ШИРОКОГО ПРИМЕНЕНИЯ

При построении систем логического управления на ИМС обычно используется следующая иерархия схемно-конструктивных единиц (блоков): корпус - модуль - субблок (кассета)-стойка - пульт.

Методы компоновки указанных блоков могут значительно различаться - от размещения полностью законченных специализированных схем до установки элементов «россыпью» с выводом всех соединений на внешние разъемы. При использовании первого метода резко возрастает номенклатура блоков, а при втором - избыточность, так как обычно имеет место ограничение по числу внешних выводов. Кроме того, при втором методе остается неясным: какими элементами и в какой пропорции заполнить эти блоки?

Ответ на этот вопрос зависит от структуры рассматриваемых схем. В дискретных системах обычно выделяют группы регулярных и нерегулярных схем. Существенным отличием схем логического управления является то, что они в подавляющем большинстве случаев являются нерегулярными. Кроме того, специфической особенностью систем логического управления, как показано в гл. 1, является то, что ИМС, имеющие конечное число состояний, совмещают в себе функции элементов памяти, и поэтому основная проблема при решении задачи унификации сводится к унификации нерегулярных комбинационных схем.

В известных работах [31, 45], посвященных унификации таких схем, оптимальное решение определяется десятками и сотнями типов блоков. Столь широкая номенклатура блоков объясняется тем, что обычно задачи синтеза схем и их разбиения решались последовательно и независимо.

Действительно, традиционное решение задачи проектирования нерегулярных комбинационных схем обычно состоит из двух основных этапов: а) логического проектирования, в ходе которого разрабатывается принципиальная логическая схема в базисе вентилях, входящих в состав используемых ИМС; б) технического проектирования, при котором осуществляется компоновка построенной схемы в корпусах, на платах, в субблоках.

Такая стратегия построения схем сильно затрудняет проблему их схемно-конструктивной унификации и получения оценок сложности предстоящей реализации булевых формул в базисе унифицированных плат и субблоков.

Ниже излагается подход к проектированию нерегулярных комбинационных схем, при котором часть или все задачи технического проектирования решаются до или в ходе этапа логического проектирования, что позволяет устранить указанные выше ограничения.

Для устранения противоречия между избыточностью и номенклатурой предлагается использовать принцип многофункциональности, который в данном случае состоит в том, что из ИМС, серийно выпускаемых промышленностью, формируются НЛМ, которые и размещаются на плате или в субблоке.

Так как формирование модулей осуществляется по критерию минимума числа внешних выводов при заданных функциональных возможностях и модули реализуют формулы из сравнительно большого числа букв, объем внешнего монтажа резко сокращается при одновременном обеспечении широких функциональных возможностей унифицированной схемно-конструктивной единицы. Формирование унифицированных микросборок. Известно, что в настоящее время при построении устройств логического управления наиболее часто используются ИМС транзисторно-транзисторной логики потенциального типа (серии 133, 130, 136). Однако микросхемы этих серий (типа комбинационных схем) обладают сравнительно низким уровнем схемной интеграции, что приводит к большому объему неупорядоченного внешнего монтажа и к значительной трудоемкости его выполнения, а также к использованию сложных многослойных печатных плат.

С целью устранения указанных недостатков может быть выполнена разработка унифицированной системы сложных гибридных микросхем (микросборок) на базе ИМС указанных серий.

Предположим, что микросборки изготавливаются в трех конструктивных модификациях, отличающихся числом комплектующих ИМС (8; 4 и 2 микросхемы). Эти конструктивные модификации, условно названные большими (БИС), средними (СИС) и малыми интегральными схемами (МИС), имеют соответственно 34, 20 и 14 внешних выводов.

Если повышение уровня схемной интеграции регулярных схем при разработке этой системы осуществляется относительно просто, на основании общих методов теории автоматов, то проектирование унифицированных микросборок для реализации нерегулярных комбинационных схем вызывает серьезные трудности. Для их устранения было предложено сформировать по крайней мере одну микросборку системы из НЛМ, порождающие функции и таблицы настроек которых приведены в гл. 3.

При этом решаемая задача может быть сформулирована следующим образом: выбрать такие порождающие функции НЛМ, при реализации которых в базисе ИМС используемых серий удастся согласовать вместимость микросборки по числу микросхем и числу внешних выводов и обеспечить ее высокую логическую эффективность.

В ходе исследования было установлено, что удовлетворить этим требованиям удастся при использовании в составе микросборок НЛМ, универсальных в классе формул в базисе $\{\&, \vee, \bar{\quad}\}$ из трех букв.

ПФ этого модуля имеет вид

$$F(z_1, \dots, z_4) = (z_1 \vee z_2 \vee z_3)z_4 \vee z_1z_2z_3 = z_1z_4 \vee z_2z_4 \vee z_3z_4 \vee z_1z_2z_3.$$

Для реализации этой функции (с точностью до инверсии) требуется всего лишь один корпус ИМС серии 133 - микросхема К1ЛР333, структура которой описывается функцией

$$y(x_1, \dots, x_9) = x_1x_2 \vee x_3x_4 \vee x_5x_6 \vee x_7x_8x_9,$$

так как при

$$\begin{array}{lll} x_1 = z_1; & x_4 = z_4; & x_7 = z_1; \\ x_2 = z_4; & x_5 = z_3; & x_8 = z_2; \\ x_3 = z_2; & x_6 = z_4; & x_9 = z_3; \end{array}$$

справедливо соотношение

$$y(z_1, \dots, z_9) = z_1z_4 \vee z_2z_4 \vee z_3z_4 \vee z_1z_2z_3.$$

Разработанный модуль (рис. 5-2) путем настройки реализует представителей всех PN -типов бесповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из трех и менее букв:

$$\begin{array}{ll} \text{при } z_4 = 0 & y_1 = \bar{z}_1 \vee \bar{z}_2 \vee \bar{z}_3; & \text{при } z_4 = 0, z_3 = 1 & y_5 = \bar{z}_1 \vee \bar{z}_2; \\ \text{при } z_4 = 1 & y_1 = \bar{z}_1 \bar{z}_2 \bar{z}_3; & \text{при } z_4 = 1, z_3 = 0 & y_6 = \bar{z}_1 \bar{z}_2; \end{array}$$

$$\text{при } z_3 = 0 \quad y_1 = \bar{z}_1 \bar{z}_2 \vee \bar{z}_4;$$

$$\text{при } z_4 = 0, z_3 = z_2 = 1 \quad y_7 = \bar{z}_1.$$

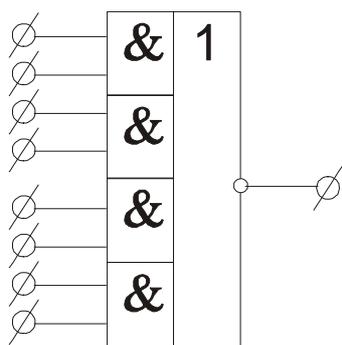
$$\text{при } z_3 = 1 \quad y_1 = (\bar{z}_1 \vee \bar{z}_2) \bar{z}_4;$$

Число таких модулей, требующаяся для реализации произвольной формулы в базисе $\{\&, \vee, \bar{}\}$ из h букв, удовлетворяет неравенству

$$\left\lceil \frac{h-1}{2} \right\rceil \leq L_1 \leq \left\lceil \frac{2(h-1)}{3} \right\rceil. \quad (5-8)$$

Покажем, что использование таких модулей в составе микросборок позволяет повысить функциональные возможности этих

А)



Б)

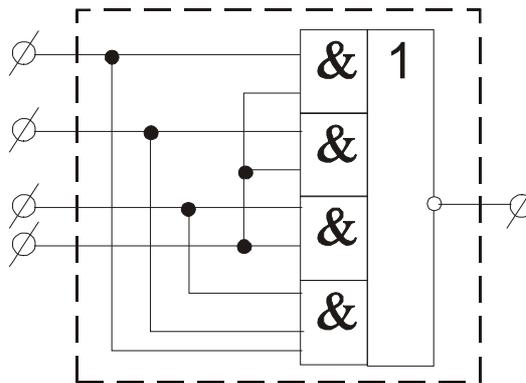


Рис. 5-2. Реализация настраиваемого логического модуля: а-микросхема К1ЛР333; б - настраиваемый логический модуль

микросборок по сравнению с применением в них непосредственно микросхем К.1ЛР333.

В § 5-1 для микросхемы К1ЛР333 найдено значение K_3 , которое равно 4,46, и поэтому для нее справедливо соотношение

$$\left\lceil \frac{h-1}{3,36} \right\rceil \leq L_2 \leq \left\lceil \frac{2(h-1)}{4,46} \right\rceil. \quad (5-9)$$

Учитывая при этом, что число внешних выводов этой ИМС $m_2 = 10$, а число внешних выводов предложенного модуля $m_1 = 5$, можно утверждать, что ожидаемый выигрыш от использования разработанных модулей в составе микросборок определяется на основе следующего соотношения:

$$\varphi = \frac{m_2}{m_1} \frac{L_1}{L_2} = \frac{10}{5} \left(\frac{2}{3,36} \div \frac{3}{4,46} \right) = 1,19 \div 1,34. \quad (5-10)$$

Применение предлагаемых модулей в составе микросборок среднего уровня интеграции позволяет согласовать между собой их вместимость по числу используемых ИМС и внешних выводов и обеспечивает высокую логическую эффективность микросборок при реализации нерегулярных комбинационных схем.

Действительно, так как в микросборке СИС, имеющей 20 внешних выводов, могут быть размещены четыре ИМС, то при установке в ней трех не связанных между собой модулей, рассмотренных выше, удастся задействовать 17 внешних выводов (3X5 плюс 2 для подачи питания) и три ИМС. Оставшиеся три вывода и одну ИМС (К1ЛБ333) удастся использовать, если вывести для каждого модуля по второму (инверсному) выходу (рис. 5-3). Наличие двух выходов (прямого и инверсного) у каждого модуля позволяет при неравнодоступности выходов *ИИИ* в ряде случаев уменьшить необходимое число инверторов.

Число таких микросборок, требующихся для реализации произвольной формулы в базисе $\{\&, \vee, \bar{}\}$ из h букв и при использовании *PN*-классификации, удовлетворяет неравенству

$$\left\lceil \frac{h-1}{6} \right\rceil \leq L_M \leq \left\lceil \frac{2(h-1)}{9} \right\rceil. \quad (5-11)$$

Рациональное использование микросборок этого типа обеспечивается на основе метода синтеза, изложенного в гл. 4. Подход изложенный выше, распространяется и на схемно-конструктивные единицы более высокого уровня интеграции по сравнению с микросборками, рассмотренными выше. Формирование унифицированных плат. Пусть имеется плата размером 90 X 90 мм с 54 внешними выводами, два из которых используются для подачи питания. На этой плате могут быть размещены 12 ИМС серии 133. Так как каждая ИМС имеет 12 логических входов и выходов, то суммарное число внешних логических выводов у этих ИМС равно 144.

Следовательно, решаемая задача сводится к тому, чтобы уменьшить внешний монтаж в $(144/52) = 2,78$ раза и обеспечить при этом широкие функциональные возможности платы при реализации различных нерегулярных комбинационных схем. Решение этой задачи состоит в использовании НЛМ, универсальных в классе формул в базе $\{\&, \vee, \bar{}\}$. Выбор модуля, универсального в классе формул в указанном базисе из трех букв,

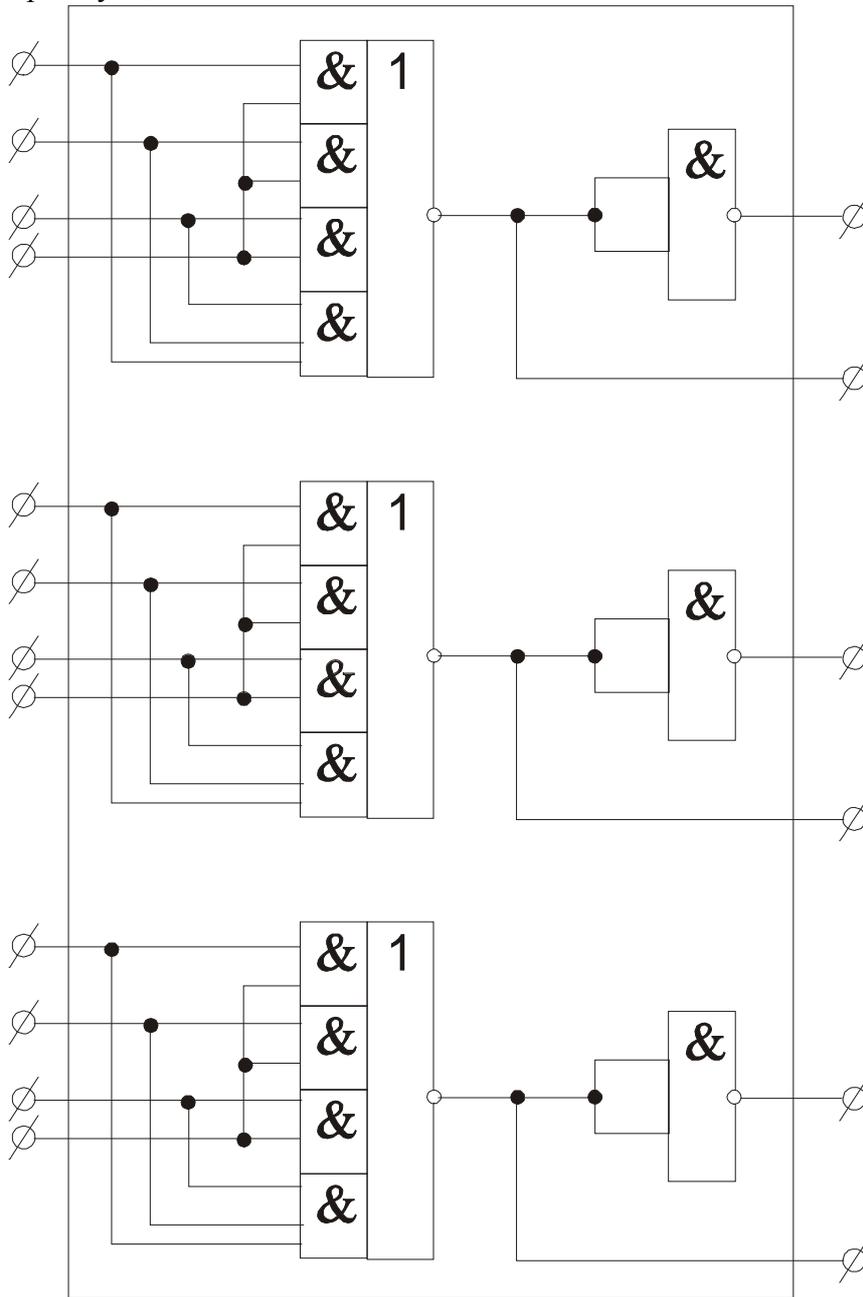


Рис. 5-3. Микросборка среднего уровня интеграции

в качестве базовой схемы позволяет удовлетворить перечисленные выше требования.

Как было показано выше, один такой модуль, реализующий при настройке положительно-монотонные формулы из трех букв, имеет пять внешних выводов и может быть построен на одной микросхеме К1ЛР333 и одной четверти микросхемы К1ЛБ333. Для реализации девяти таких модулей

потребуется девять микросхем К1ЛР333, три микросхемы К1ЛБ333 и 45 внешних выводов. Оставшиеся незадействованными три вентиля «И - НЕ» ИМС К1ЛБ333 и шесть внешних выводов используются для реализации трех инверторов. Один внешний вывод платы остается свободным и не используется.

Реализация комбинационных схем в базе таких плат также осуществляется на основе метода, изложенного в гл. 4. При этом число плат, требующихся для реализации системы из N формул, заданных в базе $\{\&, \vee, \bar{\quad}\}$, в правых частях которых содержится H букв, определяется из соотношения

$$\left\lceil \frac{\left\lceil \frac{H-N}{2} \right\rceil}{9} \right\rceil \leq L_n \leq \left\lceil \frac{\left\lceil \frac{2(H-N)}{3} \right\rceil + N - 1}{9} \right\rceil. \quad (5-12)$$

Сформированные платы устанавливаются в субблоки, имеющие разъемы для соединения с монтажным комплектом. В одном субблоке могут быть размещены восемь таких плат (по четыре с каждой стороны), которые соединяются между собой неунифицированным проводным монтажом. Таким образом, число субблоков, построенных из предложенных плат, требующихся для реализации системы из N формул, удовлетворяет неравенству

$$\left\lceil \frac{\left\lceil \frac{H-N}{2} \right\rceil}{72} \right\rceil \leq L_c \leq \left\lceil \frac{\left\lceil \frac{2(H-N)}{3} \right\rceil + N - 1}{72} \right\rceil. \quad (5-13)$$

Надо отметить, что при использовании предложенных плат унификация на уровне субблоков отсутствует. Это приводит к необходимости иметь в составе ЗИП разнотипные (по схеме соединения плат) субблоки. Дальнейшее повышение уровня унификации связано с разработкой метода построения унифицированных субблоков.

Формирование унифицированных субблоков. Предположим, что при построении систем управления применяются конструктивно-унифицированные субблоки, в каждом из которых со стороны монтажа может быть установлен разъем на 122 контакта, а с «лицевой» стороны - разъем на 61 контакт. Допустим, что в этом субблоке могут быть расположены две конструктивно-унифицированные платы, на каждой из которых могут быть размещены по 30 ИМС серии 133.

Требуется разработать унифицированную схему этого субблока, позволяющую согласовать его вместимость по числу ИМС

и внешних выводов и обеспечить при этом широкие функциональные возможности при реализации нерегулярных комбинационных схем различных систем управления, описываемых булевыми формулами в базе $\{\&, \vee, \bar{\quad}\}$. Эти формулы отличаются числом букв, а при одном и том же числе букв - структурой.

Для того чтобы при фиксированном числе букв отсутствовала зависимость схемы субблока от структуры реализуемых формул, выберем в качестве базовых схем, из которых он строится, НЛМ, универсальные для класса формул в базе $\{\&, \vee, \bar{\quad}\}$, порождающие функции которых приведены в гл. 3, а для того чтобы отсутствовала зависимость также и от числа букв, выберем однотипные модули.

Для решения сформулированной задачи применим НЛМ, у которых информационные и настроечные входы независимы. Использование таких модулей позволяет частично разгрузить разъем, связанный с монтажным комплектом, так как при этом настроечные входы модулей можно соединить с разъемом, установленным с «лицевой» стороны платы. При выборе в качестве базовой ячейки субблока модуля, универсального в классе формул в базе $\{\&, \vee, \bar{\quad}\}$ из трех букв, обладающего указанным свойством, поставленная задача может быть решена достаточно эффективно. Действительно, один такой модуль имеет три информационных и два настроечных входа, один выход и реализуется с помощью двух ИМС серии 133, и поэтому, если на каждой плате разместить по 15 таких модулей, то задача согласования будет решена, так как будут использованы все 60 ИМС, 60 выводов настроечного поля и 120 выводов разъема (без учета выводов для подачи питания).

При применении субблока, сформированного указанным способом, выполняется неравенство

$$\left[\frac{\frac{H-N}{2}}{30} \right] \leq L_{1c} \leq \left[\frac{\frac{2(H-N)}{3} + N - 1}{30} \right]. \quad (5-14)$$

Покажем, что применение настроечного поля приводит к значительному расширению функциональных возможностей субблока практически без увеличения его размеров, по сравнению с конструкцией, в которой такое поле не применяется. Предположим, что «лицевой» разъем не используется, а субблок формируется из НЛМ, универсальных в классе формул в указанном базисе из трех букв, у которых информационные и настроечные входы разделены лишь частично. Каждый такой модуль имеет четыре входа, один выход и реализуется $1\frac{1}{4}$ корпуса. Поэтому в субблоке может быть размещено лишь $(120/5) = 24$ модуля, построенных на 30 ИМС. При этом остальные 30 ИМС могут не устанавливаться, так как свободные внешние выводы на разъеме субблока отсутствуют.

Таким образом, использование предложенного подхода позволяет без увеличения размеров субблока расширить его функциональные возможности в 1,25 раза. При этом отметим, что второй способ формирования унифицированных субблоков (без применения «лицевого» разъема) обеспечивает, в свою очередь, значительно большие функциональные возможности по сравнению со способом, известным из литературы [17], при котором к разъему подсоединяются ИМС без предварительного формирования из них модулей, обладающих минимальным числом внешних выводов. В этом случае в субблоке может быть размещено лишь $(120/12) = 10$ ИМС.

ОДНОРОДНЫЕ СТРУКТУРЫ ИЗ НАСТРАИВАЕМЫХ МОДУЛЕЙ

5-1. ОДНОРОДНЫЕ СТРУКТУРЫ. ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ

Построение НЛМ из ФЭ, позволяющих при соответствующей настройке реализовать любую булеву формулу при числе букв $h \geq 6$, связано с большими вычислительными и технологическими трудностями. Поэтому реализация булевых формул из большого числа букв осуществляется обычно путем объединения модулей, универсальных в классе булевых формул из малого числа букв, в древовидные схемы на основе метода синтеза, изложенного в гл. 4.

Реализация различных формул приводит к построению схем, имеющих структурные отличия. Таким образом, межмодульные связи обычно являются неупорядоченными, что снижает технологичность изготовления систем управления.

Одним из интересных направлений повышения уровней унификации и интеграции является использование однородных структур из настраиваемых модулей. Авторы не ставят своей целью подробно ознакомить читателя с теорией и практикой построения таких структур: это сделано в обширной и содержательной литературе, изданной ранее [19, 32, 38]. Однако, занимаясь настраиваемыми модулями, мы интересовались, как получаемые результаты согласуются с достижениями в области однородных структур. К сожалению, настраиваемые модули, предложенные в гл. 3, оказались не приспособленными для их использования в этом направлении.

С другой стороны, из литературы неизвестны однородные структуры из комбинационных элементов, позволяющие реализовать булевы функции, заданные в скобочных формах произвольной глубины, которые обычно используются при записи алгоритмов работы управляющих логических устройств. Использование в однородных структурах ячеек из элементов с памятью для реализации скобочных формул произвольной глубины [8] значительно повышает сложность таких структур и осложняет их применение на практике. Поэтому мы рассмотрим вопросы разработки и применения однородных структур из настраиваемых комбинационных модулей, позволяющих реализовать формулы указанного вида.

Структура называется однородной, если она состоит из одинаковых модулей, соединенных друг с другом одинаковым образом.

Однородные структуры могут быть классифицированы по следующим основным признакам [38]:

1. По направленности связей: а) односторонние структуры, в которых подача выходного сигнала i -го модуля на вход j -го модуля исключает связь j -го модуля со входами i -го модуля; б) двусторонние структуры, у которых подача выходного сигнала j -го модуля на вход i -го модуля требует подачи выходного сигнала j -го модуля на вход i -го модуля.
2. По размерности пространства: а) одномерные (линейные), обычно называемые цепочками; б) двумерные (плоскостные), обычно называемые решетками; в) трехмерные (объемные).
3. По типу используемых модулей: а) структуры из модулей с памятью; б) структуры из комбинационных модулей.
4. По расположению модулей в структуре: а) вершинные, когда модули располагаются в вершинах соответствующего графа-структуры из ФЭ; б) реберные, когда модули располагаются на ребрах соответствующего графа - структуры из элементов с двусторонней проводимостью (ЭДП). Все входы и выходы модулей можно разделить на межмодульные (в цепочках - боковые) и внешние. На боковые входы модулей поступают сигналы с выходов других, непосредственно связанных модулей. В некоторых случаях удобно пользоваться термином «межмодульный (боковой) канал», под которым понимать связь входа данного модуля с выходом соседнего модуля.
5. По числу каналов структуры делятся на: а) одноканальные.
б) многоканальные.

Все остальные входы модулей за исключением межмодульных считаются внешними, которые в свою очередь делятся на:

- а) информационные, на которые подаются входные переменные;
 б) настроечные, на которые подаются сигналы настройки, определяющие режим работы модуля структуры.

6-2. РЕАЛИЗАЦИЯ БУЛЕВЫХ ФОРМУЛ В ОДНОРОДНЫХ СТРУКТУРАХ ИЗ КОМБИНАЦИОННЫХ ЭЛЕМЕНТОВ

Одномерная однонаправленная однородная структура из двухвходовых элементов $\&$ и \vee называется каскадом Макхападхая [68].

Каскады Макхападхая реализуют неповторные пороговые формулы, и только их. Действительно, известно, что если формула $\varphi(x_1, \dots, x_{h-1})$ неповторна, то неповторна также и формула

$$f(x_1, \dots, x_h) = \varphi(x_1, \dots, x_{h-1}) * x_h, \quad (6-1)$$

где $*$ = $\{\&, \vee\}$.

В свою очередь, известно также, что если функция $\varphi(x_1, \dots, x_{h-1})$ является пороговой, то пороговой является также и функция, задаваемая выражением (6-1) [7]. Таким образом, можно утверждать, что если $\varphi(x_1, \dots, x_{h-1})$ - неповторная пороговая формула, то неповторной пороговой является также и формула вида (6-1).

Из сказанного следует, что произвольная неповторная пороговая формула может быть представлена в виде

$$f(x_1, \dots, x_h) = (((x_1 * x_2) * x_3) \dots * x_{h-2}) * x_{h-1} * x_h. \quad (6-2)$$

В работе [68] показано, что каскады Макхападхая из $h-1$ элементов реализуют формулы из h букв, которые могут быть представлены выражением (6-2), и только их. Следовательно, справедливо утверждение: каскады Макхападхая реализуют неповторные пороговые формулы.

Покажем, что эти каскады реализуют формулы только этого класса. В работе [7] доказано, что непороговые функции не могут быть представлены в виде (6-2) и поэтому не реализуются каскадами рассматриваемого типа. В свою очередь, в работе [38] показано, что каскад Макхападхая с A входами не позволяет реализовать ни одну повторную формулу в базисе $\{\&, \vee, \bar{}\}$ из h букв, которая не была бы эквивалентна формуле, неповторной в том же базисе, содержащей меньшее число букв. Таким образом, утверждение доказано полностью.

Существует 2^{h-1} неизоморфных каскадов Макхападхая из $h-1$ двухвходовых элементов $\&$ и \vee . Справедливость этого утверждения следует из комбинаторного соотношения для подсчета числа размещений и свойства формул вида (6-1) порождать неизоморфные формулы из h букв при замене знака $*$ на $\&$ и \vee . *Одномерная однонаправленная однородная структура из двухвходовых элементов $\&, \vee, \oplus$ называется каскадом Майтра [67].* Число неизоморфных каскадов Майтра с h входами определяется из соотношения (2-18).

Обобщенные каскады. Пусть имеется одноканальный каскад, соединенный с выходом схемы. Назовем этот каскад каскадом 1-го уровня. Подсоединяя к его входам по одному выходу новых каскадов, образуем схему, содержащую каскады 2-го уровня. Подсоединяя, в свою очередь, к входам элементов каскадов 2-го уровня по одному выходу последующих каскадов, получим схему с каскадами 3-го уровня и т. д. По такой процедуре может быть сформирована любая древовидная схема из двухвходовых элементов.

Произвольная древовидная схема, построенная на каскадах, сформированных из двухвходовых элементов $\&$ и \vee $\{\&, \vee, \oplus\}$, называется обобщенным каскадом Макхападхая (обобщенным каскадом Майтра) [66]. Обобщенный каскад, содержащий каскады уровня n , называется n -уровневым каскадом.

Обобщенный каскад с h входами содержит не более $[h/2]$ каскадов. Очевидно, что наибольшим числом каскадов обладает обобщенный каскад, структура которого есть дерево с дихотомией (структура минимальной глубины). Эта структура обладает тем свойством, что в ней первый ярус содержит наибольшее число элементов. Число двухвходовых элементов в этом ярусе в структуре,

имеющей h входов, равно $\lceil h/2 \rceil$. Ввиду того что каждый элемент обобщенного каскада принадлежит одному и только одному каскаду, число каскадов в обобщенном каскаде не превышает $\lceil h/2 \rceil$.

Так как каждый каскад в обобщенном каскаде Макхападхя является пороговым элементом, то справедливо следующее утверждение: для реализации произвольной формулы в базисе $\{\&, \vee\}$ из h букв требуется не более $\lceil h/2 \rceil$ пороговых элементов (реализующих только неповторные пороговые функции).

Максимальная длина $l(h)$ одного каскада в обобщенном каскаде с h входами удовлетворяет неравенству

$$\lceil \log_2 h \rceil \leq l(h) \leq h-1. \quad (6-3)$$

Верхняя оценка достигается в случае, когда обобщенный каскад является линейным, а нижняя оценка соответствует структуре, обладающей дихотомией, в которой искомая величина совпадает с глубиной схемы Γ . Для этой структуры выполняется соотношение

$$1 + 2 + 4 + 8 + \dots + 2^{\Gamma-1} = h-1 \quad (6-4)$$

$$2^{\Gamma} - 1 = h-1 \quad (6-5)$$

$$\Gamma = l_{\text{ниж}} = \lceil \log_2 h \rceil. \quad (6-6)$$

неизоморфных структур обобщенных каскадов, соответствующих формулам из h букв, удовлетворяет рекуррентному

$$R_h = \begin{cases} \sum_{i=1}^{\lceil h/2 \rceil} R_i R_{h-i} & \text{при } h = 2k+1, \quad k \geq 1 \\ \sum_{i=1}^{h/2} R_i R_{h-i} - C_{R_{h/2}}^2 & \text{при } h = 2k, \quad k = 1; 2; 3; \\ \sum_{i=1}^{h/2} R_i R_{h-i} - C_{R_{h/2}}^2 & \text{при } h = 2k, \quad k \geq 4. \end{cases} \quad (6-7)$$

Значения R_h при $h = 1 \div 12$ приведены в табл. 6-1. На рис. 6-1, а в качестве примера приведены структуры обобщенных каскадов для $h = 5$.

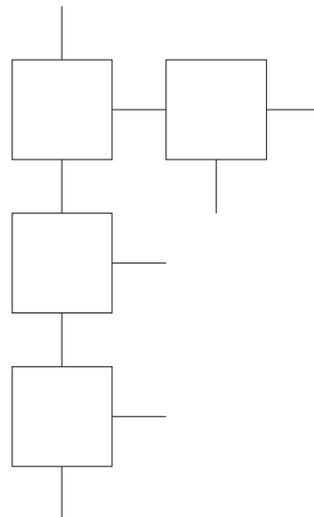
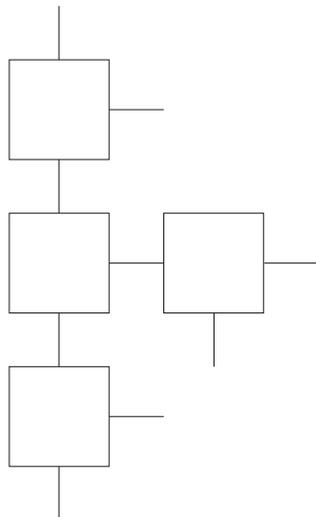
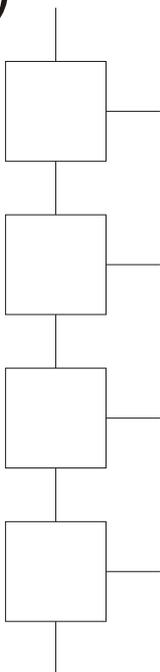
Покрывающие обобщенные каскады. В классе обобщенных каскадов наибольший интерес представляют такие обобщенные

Таблица 6-1

h	1	2	3	4	5	6	7	8	9	10	11	12
R_h	1	1	1	2	3	6	11	23	46	98	207	451

Каскады, имеющие число входов, большее h , из которых путем замены одного или нескольких элементов проводом может быть

А)



Б)

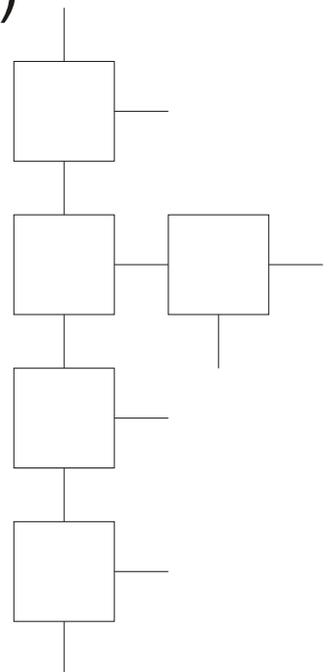


Рис. 6-1. Каскады для $h = 5$: a - обобщенные; b - покрывающий обобщенный

получен любой из обобщенных каскадов, имеющих h входов. Такие каскады являются основой для построения схем, универсальных в классе неповторных формул, обладающих простой внутренней структурой.

В табл. 6-2 приведено число элементов в покрывающих обобщенных каскадах в зависимости от числа информационных входов h .

Таблица 6-2

Число элементов в покрывающих обобщенных каскадах

h	2	3	4	5	6	7	8
$T(h)$	1	2	4	5	8	10	15

На рис. 6-1,6 в качестве примера приведена структура покрывающего обобщенного каскада для $h = 5$.

Настраиваемые каскады Макхападхая. Одномерная одноканальная однородная структура из модулей, настраиваемых на реализацию формул $y_1 = x_1 x_2 \vee (x_1 \vee x_2) z = x_1 \# x_2 \# z$. и $y_2 = x_1 \vee x_2$, называется настраиваемым каскадом Макхападхая. Модуль, реализующий при настройке указанные формулы, описывается мажоритарной функцией трех переменных, которая имеет следующий вид:

$$y = x_1 x_2 \vee (x_1 \vee x_2) z = x_1 \# x_2 \# z. \quad (6-8)$$

При этом входы x_1 (боковой) и x_2 (внешний) являются информационными, а вход z - настроечным. Настраиваемый каскад Макхападхая из $h - 1$ модулей позволяет реализовать 2^{h-1} представителей PN -типов неповторных пороговых формул из h букв. Справедливость этого утверждения следует из того, что один такой настраиваемый каскад позволяет промоделировать любой из 2^{h-1} неизоморфных каскадов Макхападхая, что вытекает из принципов его построения. Настраиваемые каскады Макхападхая реализуют лишь те булевы формулы в базисе $\{\&, \vee\}$, которые могут быть представлены схемой с линейной структурой.

Рассмотренные каскады позволяют реализовать любую неповторную пороговую формулу при условии, что прямые и инверсные выходы источников информации равнодоступны. В случае, пи выходы источников неравнодоступны для обеспечения уникальности в указанном классе формул, одноканальная однородная структура должна быть построена из модулей, реализующих при настройке следующие шесть формул:

$$y_1 = x_1 x_2; y_2 = x_1 \bar{x}_2; y_3 = \bar{x}_1 \bar{x}_2; y_4 = x_1 \vee x_2; y_5 = x_1 \vee \bar{x}_2; y_6 = \bar{x}_1 \vee \bar{x}_2.$$

Настраиваемые каскады Майтра. Пусть требуется построить дуль, который при настройке реализует три формулы из двух букв:

$$y_1 = x_1 x_2; y_2 = x_1 \vee x_2; y_3 = x_1 \oplus x_2.$$

Построим на его основе линейную однородную структуру. Предположим, что информационные и настроечные входы этого модуля независимы. Такой модуль имеет два информационных x_1 и x_2 и два настроечных - z_1 и z_2 , а его структура опирается функцией четырех переменных

$$y = (x_1 \vee x_2) \bar{z}_1 z_2 \vee x_1 x_2 z_1 \bar{z}_2 \vee (x_1 \oplus x_2) z_1 z_2 = [(x_1 \vee x_2) \bar{z}_1 \vee x_1 \oplus x_2] z_2 \vee x_1 x_2 z_1 \bar{z}_2. \quad (6-9)$$

Приведем настройки, при которых модуль реализует заданные формулы:

$$\begin{aligned} \text{при } z_1 = 1, \quad z_2 = 0 \quad y_1 &= x_1 x_2; \\ \text{при } z_1 = 0, \quad z_2 = 1 \quad y_2 &= x_1 \vee x_2; \\ \text{при } z_1 = 1, \quad z_2 = 1 \quad y_3 &= x_1 \oplus x_2. \end{aligned}$$

Объединяя выход y предыдущего модуля с боковым входом дующего модуля, получим одномерную одноканальную однородную структуру. Одномерная одноканальная однородная структура из модулей, настраиваемых на реализацию формул $y_1 = x_1 x_2, y_2 = x_1 \vee x_2, y_3 = x_1 \oplus x_2$, называется настраиваемым каскадом Майтра.

Настраиваемые каскады Майтра реализуют лишь те булевы формулы в базисе $\{\&, \vee, \oplus\}$, которые могут быть представлены схемой с линейной структурой. Справедливость этого утверждения вытекает из принципов построения таких каскадов.

В случае, если прямые и инверсные выходы *III* неравнодоступны, однородная структура должна быть построена из модулей, реализующих при настройке восемь типов формул:

$$y_1 = x_1 x_2; y_2 = x_1 \bar{x}_2; y_3 = \bar{x}_1 \bar{x}_2; y_4 = x_1 \oplus x_2;$$

$$y_5 = x_1 \vee x_2; y_6 = x_1 \bar{x}_2; y_7 = \bar{x}_1 \vee \bar{x}_2; y_8 = x_1 \oplus \bar{x}_2.$$

Настраиваемые обобщенные каскады Макхападхая. Пусть имеется настраиваемый каскад Макхападхая. Подсоединяя к его внешним информационным выходам по одному каскаду того же типа, получим схему 2-го уровня. Поступая аналогично, можно построить настраиваемые обобщенные каскады Макхападхая для моделирования любого обобщенного каскада этого типа. При этом наибольший интерес представляют настраиваемые обобщенные каскады Макхападхая, моделирующие покрывающие обобщенные каскады, так как такие каскады универсальны для класса формул в базисе $\{\&, \vee\}$. Эти универсальные схемы, имея относительно большое число внешних выводов, обладают простой внутренней структурой. В табл. 6-3 приведены значения $M_1(K)$ - числа внешних выводов в рассматриваемых каскадах для различных значений K .

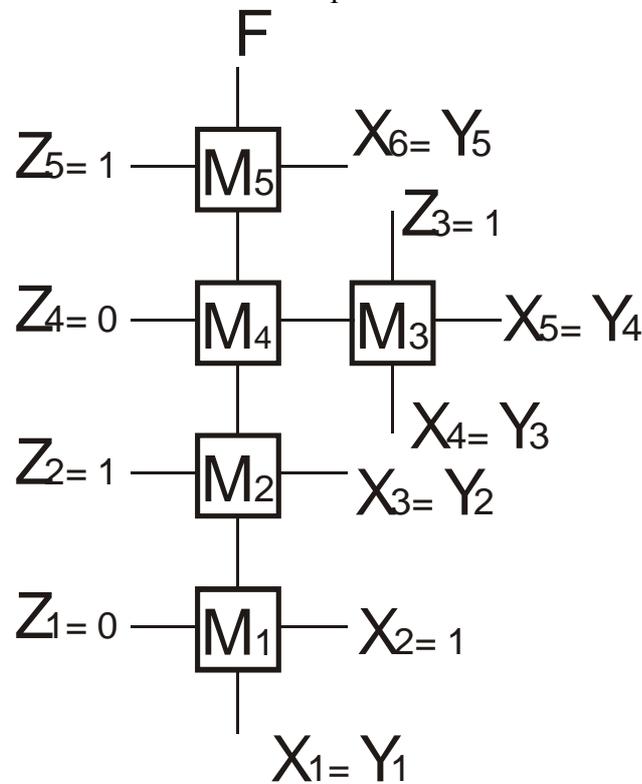


Рис. 6-2. Реализация булевой формулы $f = (y_1 \vee y_2)(y_3 \vee y_4) \vee y_5$ в настраиваемом обобщенном каскаде ($M1 \div M5$ - мажоритарные элементы)

Предложенные схемы обладают итеративной структурой, так как схема, универсальная в классе формул из h букв, может быть получена путем дополнения схемы, универсальной из $h - 1$ букв. Для таких универсальных схем можно не создавать таблицу настроек, так как настройка отдельных элементов структуры легко определяется по заданной формуле.

На рис. 6-2 в качестве примера приведена реализация формулы $f = (y_1 \vee y_2)(y_3 \vee y_4) \vee y_5$ в настраиваемом обобщенном каскаде, универсальном для класса формул в базисе $\{\&, \vee\}$ из пяти букв.

Настраиваемые обобщенные каскады Майтра. Каскады этого типа могут быть получены путем объединения настраиваемых каскадов Майтра. При этом наибольший интерес представляют настраиваемые обобщенные каскады Майтра, моделирующие покрывающие обобщенные каскады. Такие каскады универсальны для класса формул в базисе $\{\&, \vee, \oplus\}$ и обладают простой внутренней структурой. В табл. 6-3 приведены значения $M_2(K)$ - числа внешних выводов в этих каскадах для различных значений K .

Так как структура покрывающего каскада не зависит от базиса, то схемы, универсальные для класса формул в базисах $\{\&, \vee\}, \{\&, \vee, \oplus\}$,

Таблица 6-3

Число выводов в настраиваемых обобщенных каскадах

K	2	3	4	5	6	7	8
$M_1(K)$	4	6	10	12	18	22	32
$M_2(K)$	5	8	14	17	26	32	46

содержат одинаковое число модулей, соединенных одинаковым образом. При этом, однако, сами используемые модули различны, что приводит к различию в числе внешних выводов в универсальных схемах в зависимости от базиса (табл. 6-3).

Многоканальные однородные структуры. Выше было показано, что булевы формулы, которые могут быть представлены схемами с линейной структурой, реализуются в одноканальной настраиваемой структуре, ячейка которой описывается формулами вида

$$f = x * y,$$

где $*$ = $\{\&, \vee\}$ либо $*$ = $\{\&, \vee, \oplus\}$.

Для реализации булевых формул, которые могут быть представлены древовидными схемами, содержащими каскады 2-го уровня, может быть предложена двухканальная одномерная однородная структура из настраиваемых модулей. Ячейка такой структуры имеет два боковых информационных входа y_1 и y_2 , один внешний вход и два выхода f_1 и f_2 , на которых реализуются системы формул вида

$$\left. \begin{aligned} f_1 &= y_1 * x; f_1 = x; \\ f_2 &= y_2; f_2 = y_1 * y_2. \end{aligned} \right\} \quad (6-10)$$

На рис. 6-3, а приведены структурные схемы ячейки, получаемые путем настройки. При этом в случае использования базиса $\{\&, \vee\}$ число настроечных входов ячейки равно двум, а при использовании базиса $\{\&, \vee, \oplus\}$ - трем (настроечные входы на рисунке не показаны).

В первом случае структура модуля описывается следующей системой функций:

При этом, если

$$\begin{aligned} z_1 = 0, \quad z_2 = 0, \quad to \quad f_1 &= y_1 x; & f_2 &= y_2; \\ z_1 = 0, \quad z_2 = 1, \quad to \quad f_1 &= y_1 \vee x; & f_2 &= y_2; \\ z_1 = 1, \quad z_2 = 0, \quad to \quad f_1 &= x, & f_2 &= y_1 y_2; \\ z_1 = 1, \quad z_2 = 1, \quad to \quad f_1 &= x; & f_2 &= y_1 \vee y_2. \end{aligned}$$

Во втором случае структура модуля описывается системой

$$\left. \begin{aligned} f_1 &= [(y_1 \oplus x)z_2 \bar{z}_3 \vee y_1 \bar{z}_2 z_3] \bar{z}_1 \vee [z_1 (\bar{z}_2 \vee z_3) \vee \bar{z}_2 (y_1 \vee z_3)] x; \\ f_2 &= z_1 [y_1 (y_1 z_2 z_3 \vee \bar{z}_2 \bar{z}_3) \vee (y_1 \oplus y_2) z_2 z_3] \vee y_2 [\bar{z}_1 (\bar{z}_2 \vee \bar{z}_3) \vee \bar{z}_2 \bar{z}_3]. \end{aligned} \right\} \quad (6-12)$$

При этом, если

$$\begin{aligned} z_1 = 0, \quad z_2 = 0, \quad z_3 = 0, \quad to \quad f_1 &= y_1 x, & f_2 &= y_2; \\ z_1 = 0, \quad z_2 = 0, \quad z_3 = 1, \quad to \quad f_1 &= y_1 \vee x, & f_2 &= y_2; \\ z_1 = 0, \quad z_2 = 1, \quad z_3 = 0, \quad to \quad f_1 &= y_1 \oplus x, & f_2 &= y_2; \\ z_1 = 1, \quad z_2 = 1, \quad z_3 = 1, \quad to \quad f_1 &= x, & f_2 &= y_1 y_2; \\ z_1 = 1, \quad z_2 = 0, \quad z_3 = 0, \quad to \quad f_1 &= x, & f_2 &= y_1 \vee y_2; \\ z_1 = 1, \quad z_2 = 0, \quad z_3 = 1, \quad to \quad f_1 &= x, & f_2 &= y_1 \oplus y_2. \end{aligned}$$

Вложение 2-уровневого обобщенного каскада в двухканальную однородную структуру.

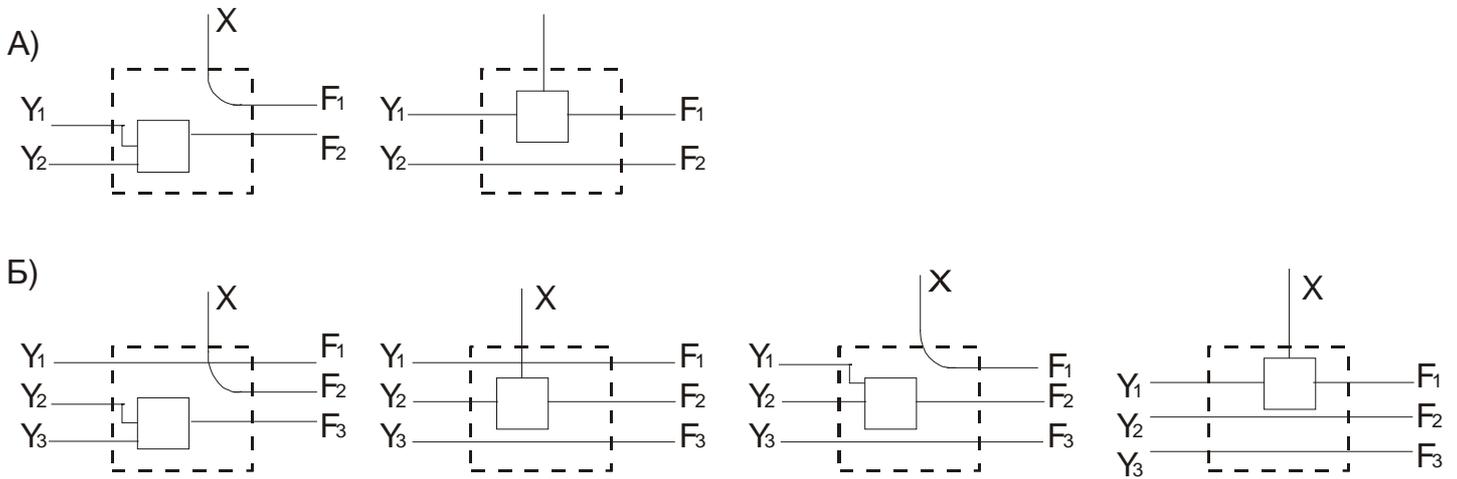


Рис. 6-3. Структурные схемы, получаемые путем настройки k -канальной ячейки: $a - k = 2; b - k = 3$

Вложение в этом случае выполняется таким образом, что: а) каждый двухвходовой элемент каскада вкладывается в одну и только одну ячейку однородной структуры; б) каждая ячейка структуры настраивается независимо; в) каждая ячейка настраивается на выполнение операции, реализуемой вкладываемым элементом; г) элементы каскада 1-го уровня размещаются в нижнем канале, а элементы каскадов

2-го уровня - в верхнем канале; д) каскады 2-го уровня вкладываются в соответствии с их нумерацией, начиная с каскада, помеченного наименьшим номером (каскады этого уровня нумеруются в порядке их подсоединения к каскаду 1-го уровня, двигаясь от его входа к выходу).

Вложение рассматриваемого обобщенного каскада начинается с вложения в нижний канал структуры всех элементов каскада 1-го уровня от первого его элемента, оба входа которого соединены с источниками информации, до элемента, связанного

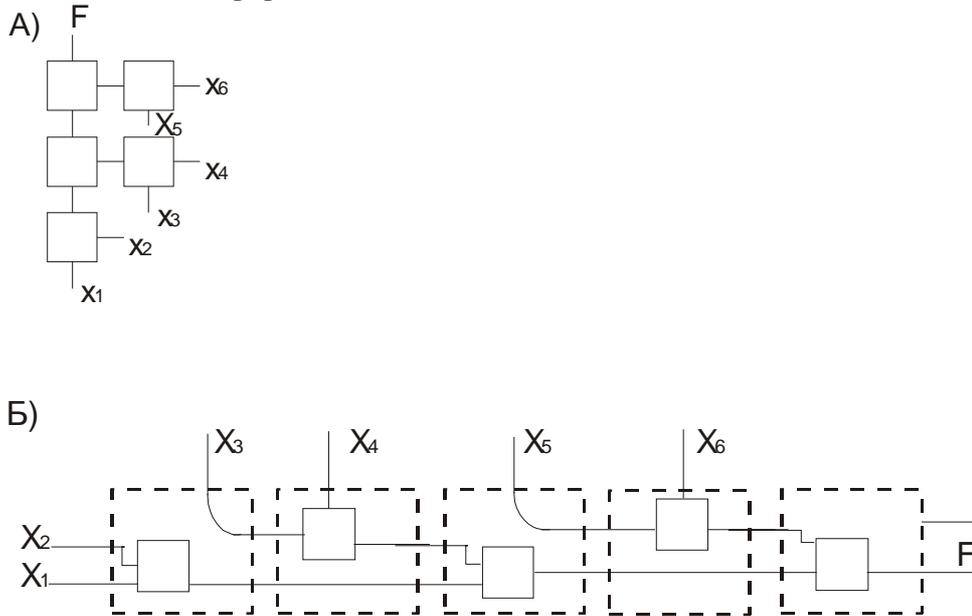


Рис. 6-4. Пример реализации булевой формулы в двухканальной однородной структуре: а - обобщенный каскад, соответствующий реализуемой формуле; б - вложение обобщенного каскада в однородную структуру

с каскадом 2-го уровня, помеченным наименьшим номером. Вложение этого элемента откладывается до тех пор, пока в верхний канал структуры не будут вложены все элементы рассматриваемого каскада 2-го уровня. Вложение элементов этого каскада осуществляется последовательно, начиная с элемента, оба входа которого соединены с источниками информации. После вложения последнего элемента указанного каскада дальнейшее вложение выполняется аналогично, начиная с элемента каскада 1-го уровня, реализация которого была отложена. На рис. 6-4 приведен пример реализации формулы $f = (x_1 x_2 \vee x_3 x_4)(x_5 \vee x_6)$ в двухканальной структуре.

Произвольная ДНФ из h букв может быть реализована в канальной структуре из $h-1$ ячеек рассматриваемого типа условия, что прямые и инверсные выходы *ИИ* равнодоступны. Справедливость этого утверждения следует из того, что любая ДНФ может быть представлена обобщенным каскадом 2-го уровня.

Многоканальные структуры. 3-уровневый обобщенный каскад с h входами может быть реализован в трехканальной структуре из $h-1$ ячеек, каждая из которых может быть настроена на реализацию следующих систем функций на боковых выходах:

$$\left. \begin{array}{llll} f_1 = y_1; & f_1 = y_1; & f_1 = x; & f_1 = x * y_1; \\ f_2 = x; & f_2 = x * y_2; & f_2 = y_1 * y_2; & f_2 = y_2; \\ f_3 = y_2 * y_3; & f_3 = y_3; & f_3 = y_3; & f_3 = y_3. \end{array} \right\} \quad (6-13)$$

На рис. 6-3, б приведены структурные схемы ячейки для указанных режимов настройки.

Авторами установлено, что для любого K -уровневого обобщенного каскада с h входами может быть предложена такая ячейка, что этот каскад будет реализован в K -канальной одномерной однородной структуре из $k-1$ таких ячеек. При этом элементы, принадлежащие 1-му уровню обобщенного каскада, размещаются в K -м канале, элементы 2-го уровня - в $(K-1)$ -м канале и т. д., вплоть до элементов уровня K , размещаемых в первом канале.

Число настроек такой ячейки при $K \geq 2$ удовлетворяет выражению

$$I = 2a(K-1), \quad (6-14)$$

где a равно двум или трем в зависимости от числа двухместных операций в используемом базисе. Общее правило выбора настроек ячеек состоит в том, что для каждой пары взаимосвязанных $(i-1)$ -го и i -го каналов выбираются настройки, характерные для двухканальной структуры:

$$\left. \begin{array}{ll} f_{i-1} = x; & f_{i-1} = y_{i-1} * x; \\ f_i = y_i * y_{i-1}; & f_i = y_i, \end{array} \right\} \quad (6-15)$$

a в остальных каналах при этих настройках выполняется функция передачи сигнала с j -го бокового входа на j -й боковой выход.

Ниже без доказательства приводится ряд утверждений, характерных для структур рассматриваемого класса.

Число уровней в обобщенном каскаде с h входами не превосходит $\lceil \log_2 h \rceil$. При этом верхняя оценка числа уровней достигается в обобщенном каскаде минимальной глубины.

Для того чтобы заданную формулу реализовать в однородной структуре, имеющей минимальное число каналов, требуется: а) реализовать ее древовидной схемой максимально возможной глубины; б) выделить в построенной схеме каскады так, чтобы наибольший номер, определяющий уровень каскадов в ней, был минимален.

Произвольная формула из h букв может быть реализована в K -канальной структуре из $h-1$ ячеек, если выполняется неравенство $h \leq 2^{k+1} - 1$, т. е. $K \leq \lceil \log_2 h \rceil$.

Произвольная функция n переменных может быть реализована в однородной структуре рассматриваемого типа, число каналов в которой не превосходит n , а число ячеек не превосходит $3(2^{n-1} - 1)$. Справедливость этого утверждения следует из соотношения

$$h \leq 3 \cdot 2^{n-1} - 2 \quad (6-16)$$

С. В. Яблонским в работе [58].

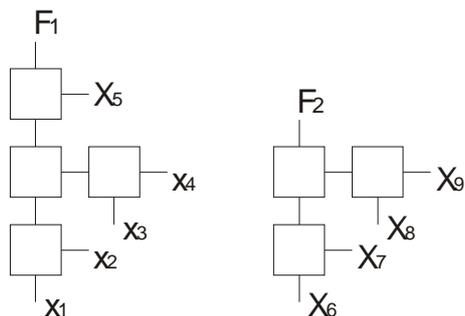
Асимптотическая оценка числа каналов при реализации функции n переменных равна $\lceil n - \log_2 \log_2 n \rceil$, а

оценка числа ячеек $\frac{2^n}{\log_2 n}$. Справедливость этого утверждения следует из соотношения

$$h = \frac{2^n}{\log_2 n}, \quad (6-17)$$

Мученного О. Б. Лупановым в работе [29]. Система из N формул, суммарное число букв в правых частях равно H , может быть реализована в одномерной однородной

А)



Б)

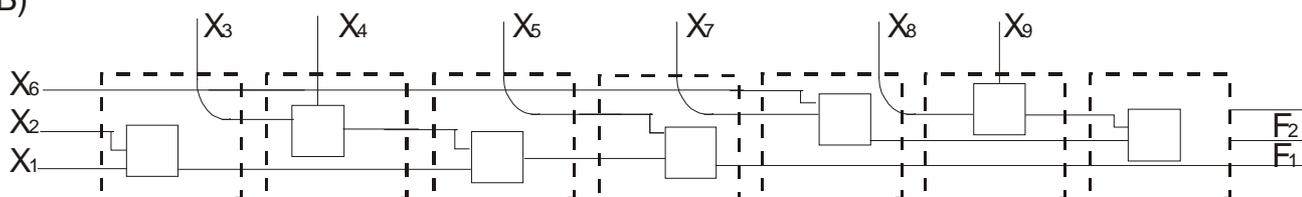


Рис. 6-5. Пример реализации системы булевых формул в трехканальной однородной структуре: а - обобщенные каскады, соответствующие реализуемым формулам; б - вложение обобщенных каскадов в однородную структуру

структуре из $H - N$ ячеек, число каналов в которой удовлетворяет соотношению

$$N \leq K \leq [\log_2 h_{\max}] + N - 1, \quad (6-18)$$

где h_{\max} - максимальное число букв в одной формуле системы.

На рис. 6-5 приведен пример реализации системы формул вида

$$\left. \begin{aligned} f_1 &= (x_1 x_2 \vee x_3 x_4) x_5; \\ f_2 &= (x_6 \vee x_7)(x_8 \vee x_9) \end{aligned} \right\} \quad (6-19)$$

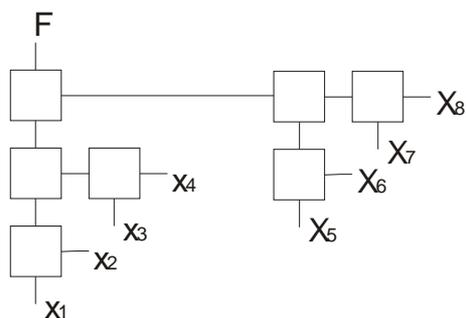
трехканальной однородной структуре.

Изложенные результаты получены в предположении, что сигналы с боковых выходов последнего элемента структуры не могут быть поданы на ее входы. Назовем такую реализацию бесповторной. Если повороты с $(K - 1)$ -го выхода структуры (за исключением крайнего нижнего выхода, на котором формируется результат) на ее входы допустимы, то класс реализуемых формул и схем в K -канальных структурах расширяется. Отметим, что при выполнении указанных поворотов структура также остается комбинационной.

Если обобщенный каскад содержит не более $K - 1$ каскадов $(K + 1)$ -го уровня, то она может быть реализована в K -канальной структуре с поворотами.

На рис. 6-6 приведен пример построения формулы $y = (x_1 x_2 \vee x_3 x_4)(x_5 x_6 \vee x_7 x_8)$, которая не может быть выполнена

A)



Б)

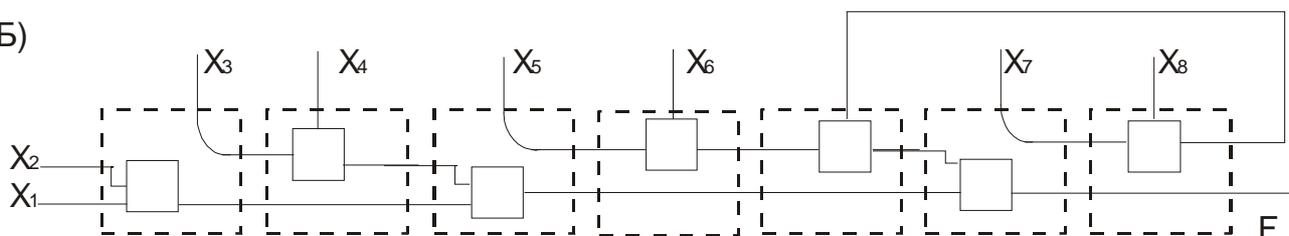


Рис. 6-6. Пример поворотной реализации булевой формулы в двухканальной структуре:

a - обобщенный каскад, соответствующий реализуемой формуле; *б* - вложение обобщенного каскада в однородную структуру в двухканальной структуре при бесповоротной реализации, но реализуется в этой структуре при выполнении поворотов.

При использовании реализации с поворотами изложенные выше результаты видоизменяются следующим образом:

$$h \leq 2^{k+2} - 5; \quad (6-20)$$

$$K \leq [\log_2 h] - 1; \quad (6-21)$$

$$K \leq n - 1; \quad (6-22)$$

$$K \leq [n - \log_2 n \log_2 n] - 1. \quad (6-23)$$

НАСТРАИВАЕМЫЕ МОДУЛИ ИЗ ЭЛЕМЕНТОВ С ДВУСТОРОННЕЙ ПРОВОДИМОСТЬЮ

7-1. ОСОБЕННОСТИ СХЕМ ИЗ ЭЛЕМЕНТОВ. С ДВУСТОРОННЕЙ ПРОВОДИМОСТЬЮ

В предыдущих главах были рассмотрены НЛМ из ФЭ, обладающих односторонней проводимостью. В данной главе исследуется вопрос о применении элементов с двусторонней проводимостью при построении и использовании НЛМ.

Интерес к классу элементов с двусторонней проводимостью связан не только с тем, что контактные элементы все еще являются основной элементной базой большого числа систем логического управления, используемых в промышленности, но и с тем, что с каждым годом все расширяется сфера применения магнитоуправляемых контактов ввиду их перспективности.

Схемы логического управления, построенные из ЭДП, обладают принципиальными отличиями по сравнению с аналогичными схемами из ФЭ:

а) исполнительные цепи ЭДП в замкнутом состоянии проводят ток в обоих направлениях. Это свойство наиболее ярко проявляется в диагональных элементах мостиковых схем. Таким образом, понятие о направлении распространения сигнала в схемах таких элементов не является таким же естественным, как в схемах из ФЭ, в которых сигнал распространяется от входа к выходу; б) в схемах из ЭДП входные элементы (например, обмотки реле) и исполнительные элементы (например, контакты) служат лишь для подачи логических переменных, в то время как реализация логических формул выполняется путем внешних соединений. При этом существенным является тот факт, что переменные подаются как в управляющие, так и в исполнительные цепи, что невозможно в схемах из ФЭ;

в) при использовании ФЭ реализация формул осуществляется в результате суперпозиции подформул, которая приводит к тому, что число входов в построенной схеме меньше суммарного числа входов применяемых элементов, в то время как в схемах ЭДП (при реализации булевых формул путем объединения исполнительных цепей) суммарное число входов элементов равно числу входов в реализуемой схеме, что приводит к сокращению числа элементов, требующихся для построения заданной формулы;

г) ЭДП в настоящее время обладают существенно большими габаритами по сравнению с ФЭ в микроэлектронном исполнении.

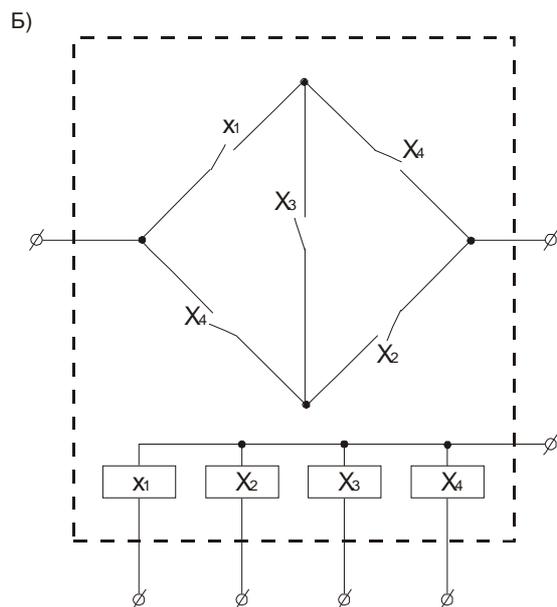
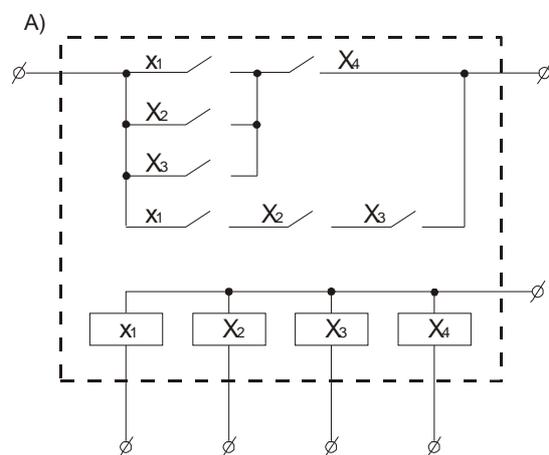
Указанные различия приводят к тому, что разработка НЛМ ЭДП должна выполняться на других принципах, чем разработка модулей из ФЭ. Так, например, при разработке модулей из релейно-контактных элементов определяющее значение имеет мнение о габаритах, и поэтому основное внимание должно уделяться выбору наиболее простой структуры; оптимизация же числа внешних выводов при этом переходит на второй план. При этом, так как избыточность по числу элементов в таких модулях должна быть минимальна или совсем отсутствовать, их настройка в основном должна осуществляться за счет избыточности по числу внешних выводов.

Таким образом, изменение критерия оптимизации требует другого подхода к построению и использованию НЛМ из ЭДП по сравнению с соответствующими методами, изложенными предыдущих разделах для модулей из ФЭ. Продемонстрируем это на примере.

В гл. 3 приведена ПФ модуля, универсального для класса формул в базисе $\{\&, \vee, \bar{}\}$ из трех букв, которая имеет следующий вид:

$$F(x_1, x_2, x_3, x_4) = (x_1 \vee x_2 \vee x_3)x_4 \vee x_1x_2x_3.$$

НЛМ, соответствующий этой ПФ, при реализации его из ФЭ, обладает минимально возможной избыточностью по числу внешних выводов, так как заданные формулы из трех букв реализуются схемой, имеющей всего лишь четыре входа.



Элементарная избыточность этого модуля не имеет практического значения ввиду предположения, что модуль из ФЭ реализуется средствами микроэлектронной интегральной технологии.

Реализуем найденную ПФ на релейно-контактных элементах. На рис. 7-1, а приведена реализация этой функции параллельно-последовательной контактной схемой, а на рис. 7-1, б – мостиковой схемой. Полученные схемы, обладая минимально возможной избыточностью по числу внешних выводов, имеют большую избыточность по числу контактов. При этом для первой схемы избыточность по числу контактов равна $\frac{7}{3}$, для второй - $\frac{5}{3}$, что ограничивает их применение на практике. Поэтому возникает необходимость в разработке нового класса НЛМ из ЭДП, обладающих минимальной элементарной избыточностью.

7-2. ВЫБОР СТРУКТУРЫ НАСТРАИВАЕМЫХ МОДУЛЕЙ ИЗ ЭДП

в настоящем параграфе рассматривается новый класс НЛМ, формирование которого основано на использовании свойств ЭДП. Это позволяет обеспечить их построение без избыточности по числу элементов или с минимальной избыточностью. Рассмотрение модулей этого класса будет выполняться на примере релейно-контактных устройств, однако результаты, полученные при этом, будут справедливы также и для других типов элементов, исполнительные цепи которых обладают двусторонней проводимостью.

В качестве основной структуры НЛМ из ЭДП предлагается конфигурация, представленная на рис. 7-2.

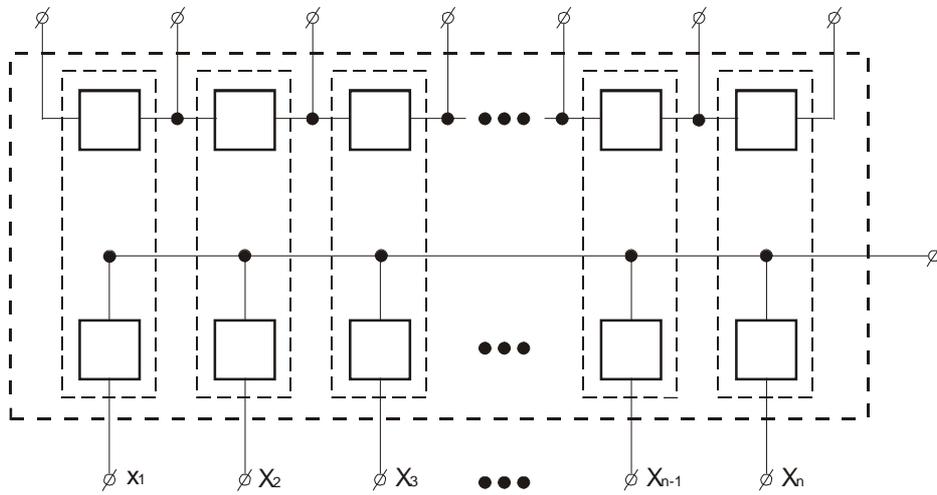


Рис. 7-2. Настраиваемый модуль типа цепочки

Эта структура образована из элементов (реле), у которых исполнительные цепи (контакты), обладающие двусторонней проводимостью, соединены последовательно, причем точки их объединения и крайние выходы структуры соединены с внешними выводами модуля. Структуру такого вида будем в дальнейшем называть цепочкой. Управляющие цепи (обмотки) либо объединяются звездой (в случае, если обмотки нескольких реле имеют общую точку, соединенную с одним из внешних выводов модуля), либо не связываются между собой. Возможны также промежуточные варианты, когда часть обмоток соединяются звездой, а остальные не связываются. Отметим, что способ соединения обмоток зависит от принятой организации «раздачи» питания (централизованное или от разных источников). Настройка таких модулей может осуществляться как со стороны входов (обмоток), так и со стороны выходов (контактов) (рис. 7-3). При этом во входных цепях в качестве операций настройки используются подача констант 0 и 1 и отождествление входов, а в выходных – отождествление выходов путем наложения перемычек. В модулях этого типа настройка выходных цепей является основной, что принципиально отличает их от одновыходных модулей из ФЭ, которые могут настраиваться только со стороны входов. Предлагаемые модули отличаются также и от многовыходных модулей из ФЭ, так как в последних настройка со стороны выхода сводится к выбору того или иного выхода и осуществляется без внешней коммутации.

Исследуем функциональные возможности предложенных модулей. Для этого первоначально рассмотрим цепочку из двух контактов. Очевидно, что она позволяет реализовать путем настройки каждого представителя PN -типов неповторных формул в базисе $\{\&, \vee, \bar{\quad}\}$ из двух букв ($y_1 = x_1x_2, y_2 = x_1 \vee x_2$). При этом для реализации первого представителя наложение перемычек не требуется, а второй реализуется с помощью одной перемычки. Дальнейшее исследование функциональных возможностей

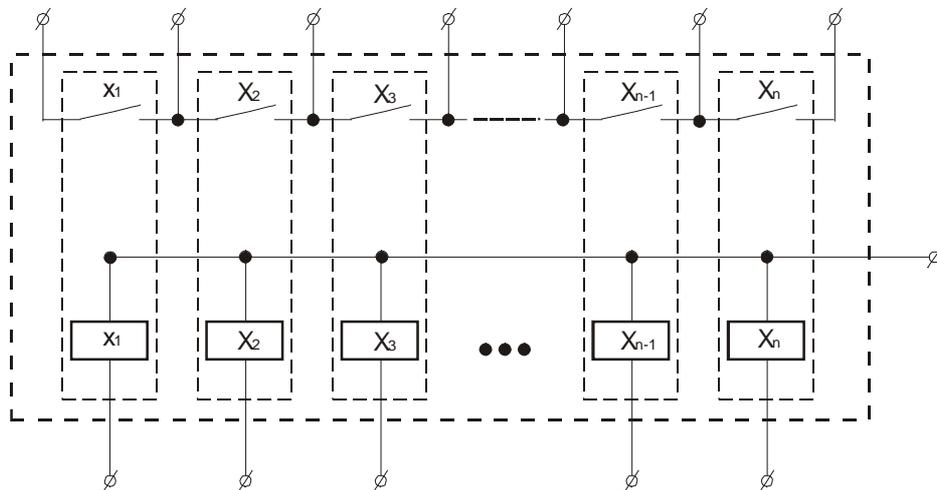


Рис. 7-3. Настраиваемый модуль из релейно-контактных элементов
 предложенных модулей показало, что цепочка из трех контактов реализует путем настройки любого из четырех представителей указанного класса формул из трех букв, цепочка из четырех контактов - любого из 10 представителей этого же класса формул из четырех букв, а цепочки из пяти и шести контактов соответственно любого из 24 и 66 представителей этого класса из пяти и шести букв.

Из сказанного следует, что предложенные модули из l контактов универсальны для класса формул в базисе $\{\&, \vee, \bar{\quad}\}$ из $t = h$ букв при $l \leq 6$. Сравнивая, например, элементную сложность такого модуля, универсального для класса формул из трех букв (три контакта), со сложностью модулей, построенных в предыдущем параграфе на основе порождающей функции (семь и пять контактов), можно сделать вывод о высокой эффективности предлагаемых модулей по этому критерию.

Выполним сопоставление предложенных модулей с контактами, используемыми «россыпью». Сравнение результатов реализации двухполюсных параллельно-последовательных схем из шести и менее контактов в базисе цепочки и в базисе контактов, используемых «россыпью», по числу внешних выводов и перемычек показало, что цепочка обеспечивает реализацию этих схем при существенно меньших значениях этих показателей [54].

Таким образом, в этом классе схем связанная цепочка контактов и контакты «россыпью», обладающие всеми степенями свободы соединений, имеют одинаковые функциональные возможности.

Так как цепочка обладает однородной структурой, а любая параллельно-последовательная схема из $d \leq 6$ контактов реализуется в цепочке из $l = d$ контактов, то можно утверждать, что для реализации указанного класса схем предложена однородная настраиваемая структура, в которой избыточность по числу элементов отсутствует даже на уровне электроэлементов.

Необходимо отметить, что это положение противоречит устоявшейся точке зрения в этой области, так как обычно утверждается, что возможность настройки, в особенности при однородности структуры, связана с избыточностью по числу элементов.

Для предложенной структуры такая избыточность возникает лишь при $h \geq 7$. При этом 179 из 180 представителей PN -типов формул в базисе $\{\&, \vee, \bar{\quad}\}$ из семи букв безызыточно реализуется путем настройки цепочки из семи контактов, но существует один тип формул: $y = [(x_1 \vee x_2)x_3 \vee (x_4 \vee x_5)x_6]x_7$, который требует для своей реализации цепочку из восьми контактов. При увеличении значения h доля формул, реализуемых цепочкой из контактов безызыточно, уменьшается. Так, при $h = 8$ из представителей 522 PN -типов формул в базисе $\{\&, \vee, \bar{\quad}\}$ цепочка из восьми контактов позволяет реализовать представителей лишь 518 типов формул, в то время как представители остальных четырех типов:

$$y_1 = [(x_1 \vee x_2)x_3 \vee (x_4 \vee x_5)x_6]x_7x_8; y_2 = [(x_1 \vee x_2)x_3x_4 \vee (x_5 \vee x_6)x_7]x_8; y_3 = [(x_1x_2 \vee x_3)x_4 \vee (x_5 \vee x_6)x_7]x_8; \\ y_4 = [(x_1 \vee x_2)x_3 \vee (x_4 \vee x_5)x_6 \vee x_7]x_8$$

- требуют для своей реализации цепочку из девяти контактов.

Возможность настройки предложенной структуры существует за счет избыточности по числу внешних выводов по сравнению с не настраиваемой схемой, подлежащей реализации. Это наглядно видно на примере реализации формулы $y = x_1x_2x_3\dots x_{l-1}x_l$. Действительно, в этом случае цепочка имеет $l + 1$ внешних выводов, в то время как не настраиваемая схема, реализующая ту же формулу, лишь два.

Из сказанного выше следует, что предложенная структура при $l \leq 6$ может рассматриваться как НЛМ, универсальный для класса формул в базисе $\{\&, \vee, \bar{\quad}\}$ из $K = l$ букв, при условии равнодоступности прямых и инверсных выходов *III*.

При неравнодоступности выходов *III* в качестве модуля, универсального для указанного класса формул при $h \leq 6$, предлагается использовать цепочечную структуру из $2h$ реле, полученную путем объединения двух цепочек, первая из которых содержит h одноконтактных реле с замыкающими контактами, а вторая - h одноконтактных реле с размыкающими контактами (рис. 7-4). Необходимо отметить, что при реализации формул из h букв в базисе такой структуры лишь на h обмоток подаются информационные сигналы, в то время как на остальные h обмоток сигнал либо не подается (константа 0), либо подается постоянный сигнал (константа 1). При этом константа 1 должна

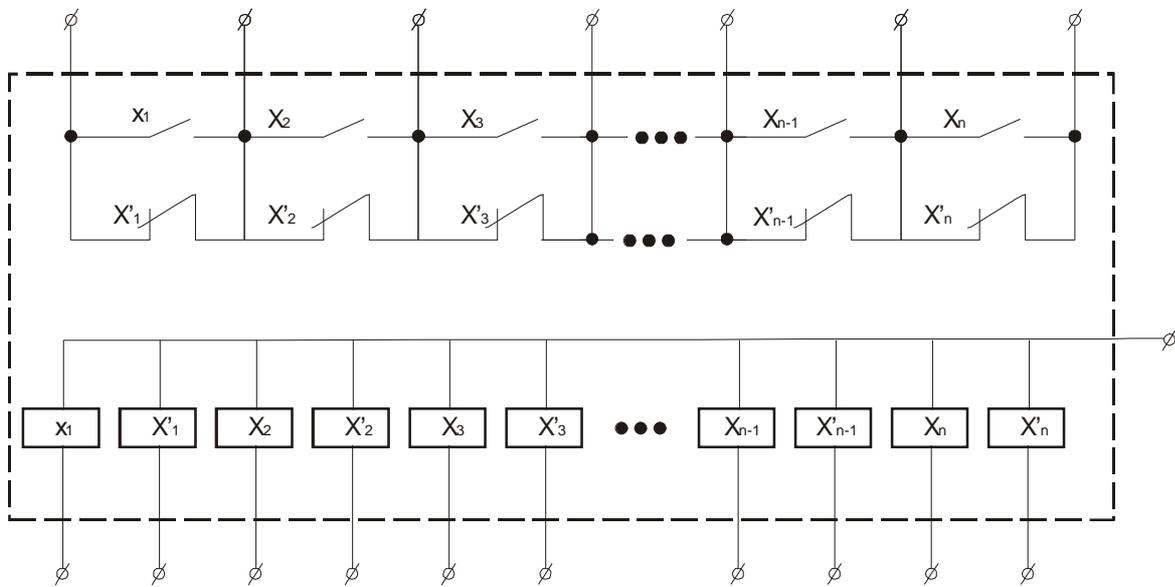


Рис. 7-4. Настраиваемый модуль из двух цепочек реле

быть подана на h' обмоток, где h' - число букв без инверсий в реализуемой формуле.

Таким образом, для предложенной структуры характерна двойная элементная избыточность при большом числе внешних выводов, что обычно неприемлемо на практике.

Меньшей избыточностью обладает структура из h реле, каждое из которых имеет по одному замыкающему и одному размыкающему

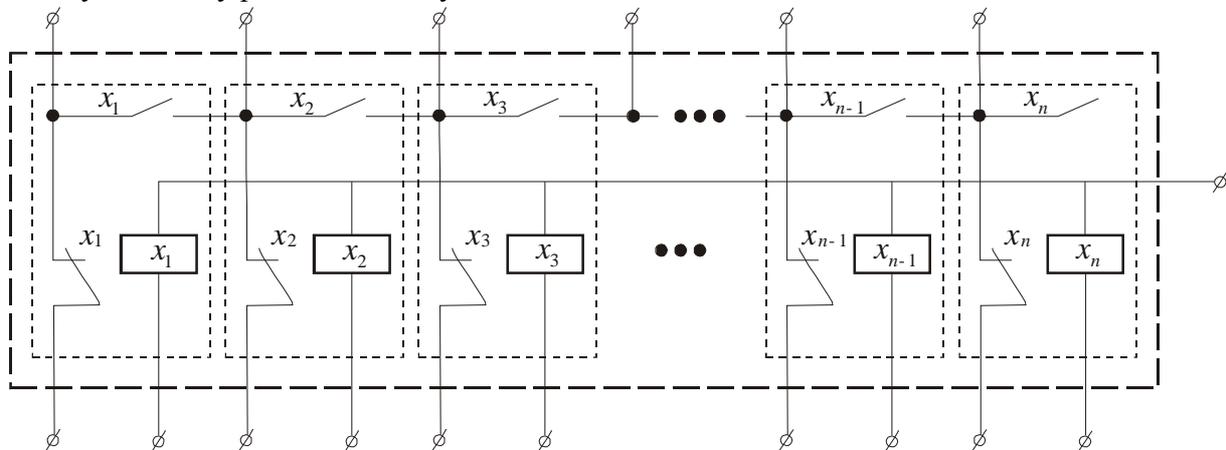


Рис. 7-5. Настраиваемый модуль из замыкающих и размыкающих контактов

контакту (либо по одному переключающему контакту), причем замыкающие контакты соединены в цепочку (рис. 7-5). Исследование функциональных возможностей такой структуры показало, что схема из двух реле позволяет путем настройки в выходных цепях реализовать любого из шести представителей P -классификации формул в базисе $\{\&, \vee, \bar{}\}$ из двух букв, а схема из трех реле - любого из 20 представителей этого класса формул из трех букв.

В заключение необходимо отметить, что предложенные модули обладают высокой контролепригодностью, так как промежуточные выводы обеспечивают возможность проверки каждого контакта в структуре в отдельности.

7-3. ПОСТРОЕНИЕ СХЕМ ИЗ ЭДП В БАЗИСЕ НАСТРАИВАЕМЫХ МОДУЛЕЙ

В гл. 4 был описан метод реализации произвольных нормальных булевых формул из h букв, заданных в булевом базисе, на основе НЛМ из ФЭ, универсальных для того же класса формул из K букв. При этом было показано, что число таких модулей в схеме удовлетворяет неравенству

$$\left\lceil \frac{h-1}{K-1} \right\rceil \leq L \leq \left\lceil \frac{2(h-1)}{K} \right\rceil. \quad (7-1)$$

В предыдущем параграфе, в свою очередь, мы установили, что модули из $l \leq 6$ ЭДП по своим функциональным возможностям эквивалентны модулям из ФЭ, универсальным для класса формул в базисе $\{\&, \vee, \bar{}\}$ из $K = l$ букв. Поэтому для реализации формул в булевом базисе из h букв $h > 1$ с помощью модулей из ЭДП может быть использован предложенный выше метод реализации. При этом для случая $l \leq 6$ справедлива оценка сложности предстоящей реализации, которая задается неравенством

$$\left\lceil \frac{h-1}{l-1} \right\rceil \leq L' \leq \left\lceil \frac{2(h-1)}{l} \right\rceil. \quad (7-2)$$

Пример. Реализовать формулу

$$y = (x_1 \vee x_2)x_3 \vee x_4x_5$$

в базисе НЛМ из трех контактов.

Ввиду того что в данном случае $h = 5$ и $l = 3$,

$$2 \leq L' \leq 3.$$

На рис. 7-6, а представлена структурная схема, реализующая данную формулу, а на рис. 7-6, б - соответствующая принципиальная схема, построенная из двух цепочек.

Несмотря на возможность применения указанного метода, ориентированного на использование функциональных возможностей модулей для класса формул в базисе $\{\&, \vee, \bar{}\}$, он в большинстве случаев не является оптимальным для схем из ЭДП ввиду их специфики. Ограниченность его объясняется тем, что:

а) при его использовании свойство двусторонней проводимости учитывается лишь в пределах каждого отдельного модуля из ЭДП на этапе настройки, а далее он выступает в качестве модуля из ФЭ. При этом реализация формул из $h > l$ букв осуществляется путем соединения входов и выходов модулей, что неэффективно для ЭДП, так как их объединение возможно путем присоединения исполнительных цепей ЭДП таким образом, чтобы суммарное число входов элементов было равно числу входов в собранной из них схеме, чего нет в схемах из ФЭ;

б) в схемах из ФЭ, если в каком-либо модуле с l входами используется лишь K входов, считая слева направо, то все $l - K$ незадействованных входов являются избыточными. В свою очередь, если применяется цепочка из l ЭДП и в ней используются лишь первые K элементов, то существует возможность изоляции этих элементов от остальных с помощью всего лишь одного элемента,

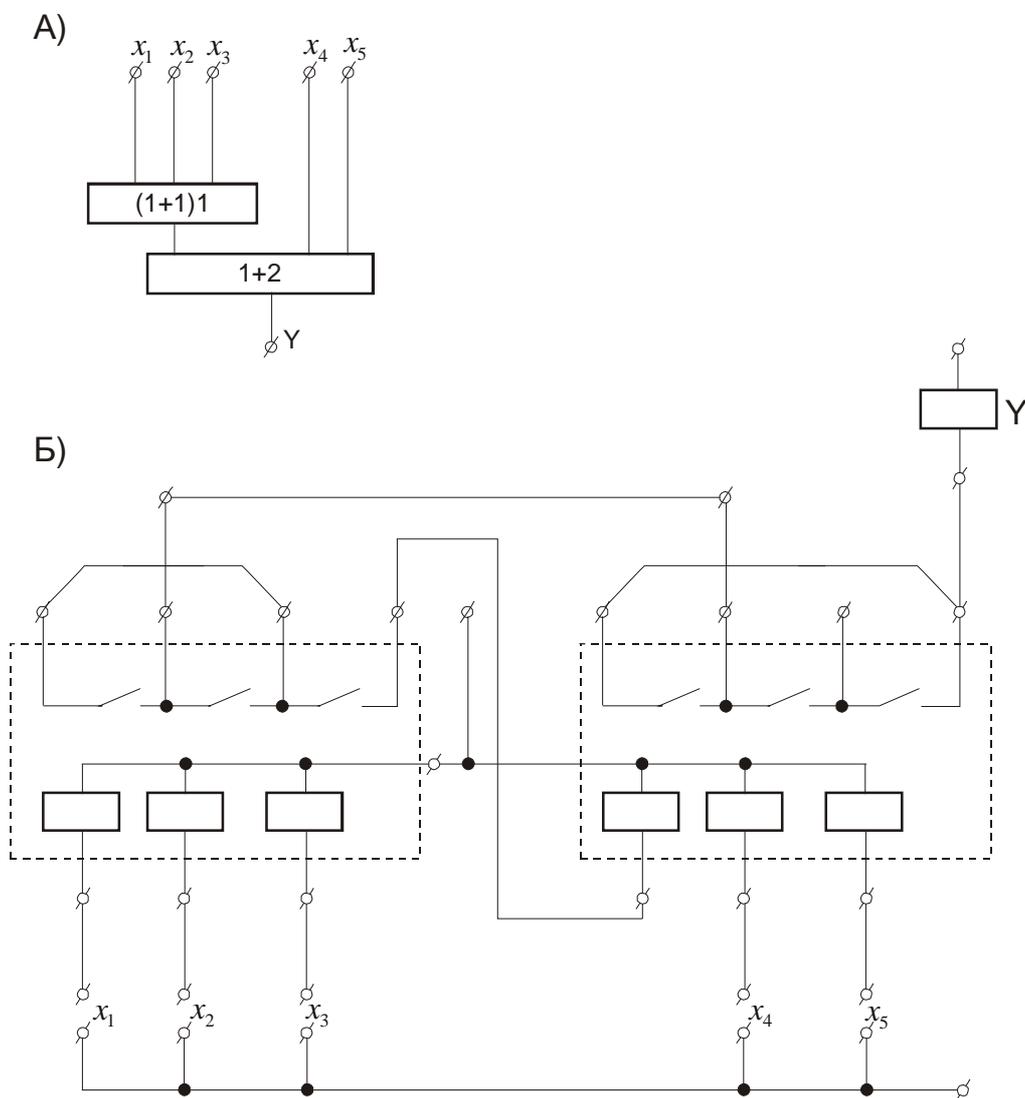


Рис. 7-6. Схема из настраиваемых модулей: а - структурная; б - контактная

являющегося избыточным; при этом остальные $l - K - 1$ ЭДП могут применяться независимо от первых K элементов;

в) предложенные в предыдущем параграфе модули из ЭДП являются многовыходными и позволяют поэтому реализовывать при настройке системы булевых формул, однако рассмотренный выше метод реализации не ориентирован на использование многовыходных модулей;

г) в ряде случаев оптимальная схема из ЭДП принадлежит к классу мостиковых схем, для которых описание в виде булевых формул неадекватно структуре схемы;

д) в схемах из ЭДП значительный процент логических операций может быть реализован непосредственно на контактах ключей управления и сигнализаторов. Это приводит к тому, что после удаления контактов этих устройств оставшаяся подсхема может обладать такой структурой, которая не описывается ни одной булевой формулой, ни системой формул.

Пример. Пусть контактная схема реализует формулу $y = (x_1 x_2 \vee x_3) x_4$ (рис. 7-7, а), причем переменная x_1 подается с помощью контакта сигнализатора. Исключим этот контакт из схемы; при этом реализации в контактном базисе подлежит конфигурация (рис. 7-7, б), которая не может быть описана на языке булевой алгебры.

Перечисленные особенности приводят к необходимости разработки нового метода рационального использования модулей из ЭДП, который должен быть ориентирован не на функциональные свойства реализуемых формул и применяемых модулей, а на топологические особенности схем, подлежащих реализации, и собственно модулей. При этом модули из ЭДП должны настраиваться не на булевы формулы, а на конфигурации реализуемых схем.

При синтезе схем на модулях из ФЭ исходными данными являлись: формула, подлежащая реализации, и набор формул, реализуемых модулями. Отметим, что исходная формула при этом

получалась традиционными методами [43], а использование модулей никак не влияло на способы ее построения, минимизации и т.д. Аналогично, при синтезе схем на модулях из ЭДП будем предполагать, что в качестве исходной задана «контактная» схема, построенная с помощью какого-либо известного метода синтеза [13]. Эту схему предстоит «покрыть» модулями, не внося в нее никаких изменений. Задачу «покрытия» рассмотрим для четырех разновидностей модулей: а) единой цепочки, длина которой теоретически не ограничена; б) совокупности цепочек длиной l' ; в) совокупности цепочек, длина которых определяется в ходе процесса реализации; г) совокупности цепочек, номенклатура которых фиксирована, суммарное число контактов равно числу контактов в реализуемой схеме.

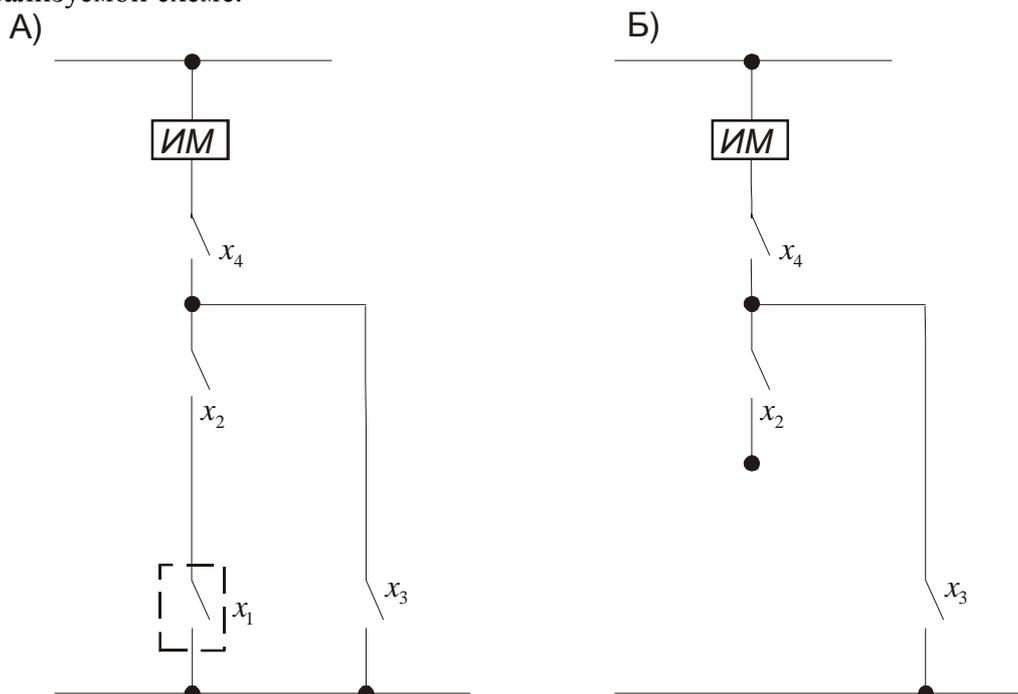


Рис. 7-7. Схема управления исполнительным механизмом (ИМ): а - с сигнализатором; б - без сигнализатора

Остановимся на решении сформулированной задачи для первой разновидности модулей. Использование остальных типов модулей базируется на результатах, получаемых для единой цепочки.

Суть излагаемого подхода в том, что при реализации схемы предлагается полностью отвлечься от свойств реализуемых функций и все внимание уделить рассмотрению топологических характеристик заданной схемы и используемых модулей. Назовем такой подход топологическим. Реализуемую схему будем рассматривать как граф, ребрами которого являются элементы исполнительных цепей (контакты), а вершинами - соединения между ними. В свою очередь, цепочку будем рассматривать как линейный граф. Внешние выходы (вершины) цепочки (линейного графа) могут взаимно соединяться с помощью внешних перемычек (дополнительных ребер), образуя при этом схемы (графы) различной конфигурации.

Отмеченные аналогии позволяют использовать некоторые результаты теории эйлеровых графов для решения задачи реализации заданной схемы с помощью единой цепочки [22]. Решение этой задачи состоит из трех основных этапов: а) «покрытия» заданной схемы цепочкой - определения эйлерова маршрута в заданной схеме; б) нумерации определенным образом выводов контактов, образующих эйлеров маршрут; в) объединения выводов цепочки в соответствии с введенной нумерацией и структурой реализуемой схемы.

Первый этап реализации, как указано выше, выполняется на основе использования свойств эйлеровых графов. Сформулируем некоторые из этих свойств применительно к контактному схематическому решению. (Доказательства этих свойств аналогичны соответствующим доказательствам, известным из теории эйлеровых графов, и поэтому в настоящем параграфе не приводятся.)

Контактную схему, которой соответствует эйлеров граф, будем называть эйлеровой, или уникальной (проходимой). Такие схемы названы уникальными ввиду того, что для них существует возможность непрерывного «прохождения» всех контактов, без повторения какого-либо из этих контактов. Поэтому уникальные схемы могут быть «пройдены» без отрыва руки от бумаги.

Последовательность контактов, «проходимых» в результате обхода уникурсальной схемы, назовем эйлеровым маршрутом.

Число узлов в схеме, образованных нечетным количеством контактов, четно. Схема является уникурсальной, если число узлов в ней, образованных нечетным числом контактов, равно 0 или 2. Если схема содержит два нечетных узла, то ее можно «пройти» только в случае, когда ее обход начинается из одного из нечетных узлов (при этом обход должен завершиться в другом нечетном узле схемы). Если нечетные узлы в схеме отсутствуют, то всегда существует возможность «пройти» схему, начиная обход из любого узла.

Выше были указаны критерии существования эйлерова маршрута в схеме. Для практического выявления этого маршрута необходим соответствующий алгоритм. Из теории графов известен алгоритм Хоанг Туи для выявления эйлеровой цепи в случае, если граф содержит две нечетные вершины, который может без изменения быть использован для определения эйлерова маршрута в контактной схеме, содержащей два нечетных узла [22].

Алгоритм Хоанг Туи. Пусть x и y - две вершины (не обязательно различные) связанного графа G . При этом все отличные от x и y вершины G обладают четными степенями, а степени вершин x и y (в случае $x \neq y$) нечетны. Применение к такому графу излагаемого ниже алгоритма позволяет всегда получить некоторую эйлерову цепь, проходящую между x и y .

1. Находим простую цепь θ , соединяющую x и y ; если ее длина больше нуля (т.е. искомая эйлерова цепь не является циклом), то все ребра цепи θ помечаем цифрой 0.

2. Если в G имеются непомеченные еще ребра, то среди них выбираем такое ребро u , которое инцидентно хотя бы одной вершине цепи θ . В подграфе, порожденном всеми непомеченными ребрами, выявляем простой цикл, содержащий ребро u , и все ребра этого цикла помечаем цифрой 1.

3. Если в G остались непомеченные ребра, то из них выбираем такое ребро v , которое смежно хотя бы с одним помеченным. В подграфе, порожденном непомеченными ребрами, выявляем простой цикл, содержащий ребро v , и все ребра этого цикла помечаем цифрой 2 и т. д. Процесс продолжается до тех пор, пока не будут помечены все ребра G .

4. После этого строим маршрут следующим образом. В качестве начальной вершины выбираем $x_0 = x$. Если уже построен маршрут $x_0 u_1 x_1 u_2 x_2 \dots x_{k-1} u_k x_k$ ($k \geq 0$), в котором все ребра u_1, u_2, \dots, u_k различны, то в случае, если $k = m$, где m - число ребер в графе G , процесс прекращаем, а в случае, если $k < m$, среди ребер, инцидентных вершине x_k и не совпадающих ни с одним из u_1, u_2, \dots, u_k , выбираем такое ребро, которое помечено наибольшей цифрой (если таких ребер несколько, то выбираем любое из них), и добавляем его к маршруту в качестве u_{k+1} , а за x_{k+1} выбираем ту вершину, с которой выбранное ребро соединяет вершину x_k (возможен случай $x_{k+1} = x_k$, когда u_{k+1} - петля).

Можно доказать, что описанный процесс действительно осуществим и результирующий маршрут будет эйлеровой цепью между x и y .

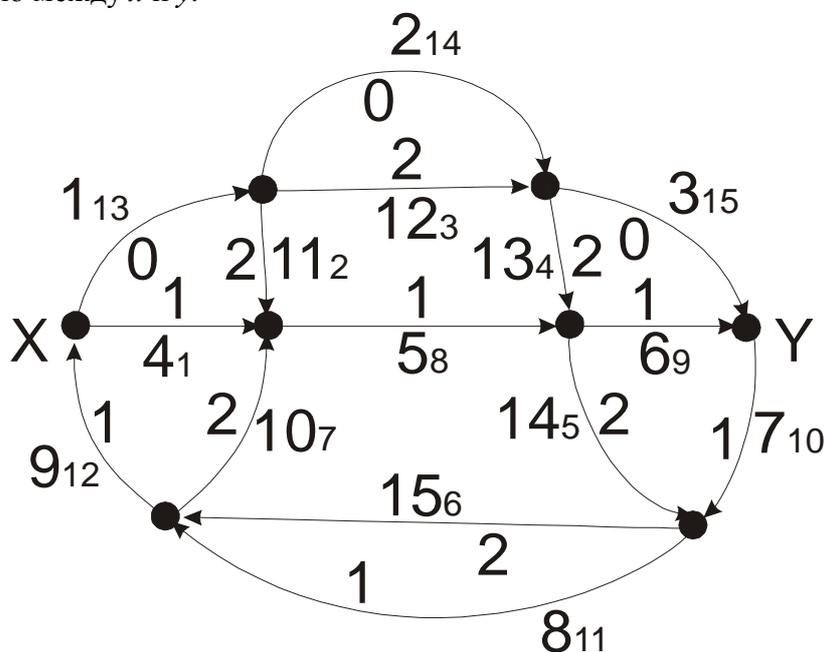


Рис. 7-8. Определение эйлеровой цепи в графе

Пример. Пользуясь алгоритмом Хоанг Туи, построить эйлерову цепь для графа, представленного на рис. 7-8.

1. В качестве простой цепи, соединяющей вершины x и y , выбираем маршрут, включающий ребра с номерами 1, 2, 3. Помечаем их значком 0.

2. Так как в графе имеются не помеченные еще ребра, то среди них выбираем ребро 4, которое инцидентно вершине x . В подграфе, порожденном непомеченными ребрами, выявляем простой цикл, содержащий ребро 4, 4-5-6-7-8-9. Все ребра этого цикла помечаем цифрой 1.

3. Так как в графе имеются непомеченные ребра, то из них выбираем ребро 10, которое смежно с помеченными ребрами. В подграфе, порожденном непомеченными ребрами, выявляем простой цикл, содержащий ребро 10, 10-11-12-13-14-15. Все ребра этого цикла помечаем цифрой 2.

4. Построение маршрута начинается из вершины x . Среди ребер, инцидентных этой вершине, выбираем ребро, помеченное наибольшей цифрой. Таким ребром является ребро 4. Для вершины, инцидентной с этим ребром, наибольшую цифру имеют ребра 10 и 11. Выбираем ребро 11 и включаем его в маршрут. Продолжая этот процесс, получим следующий маршрут: 4-11-12-13-14-15-10-5-6-7-8-9-1-2-3.

В случае, если в схеме нечетные узлы отсутствуют, используется модификация алгоритма Хоанг Туи, суть которой состоит в следующем.

В схему добавляется фиктивный контакт, располагаемый между узлом наибольшей степени, от которого начинается обход, и любым другим узлом. С помощью алгоритма Хоанг Туи определяется эйлеров маршрут. После выполнения алгоритма фиктивный контакт изымается.

Пример. Определить с помощью алгоритма Хоанг Туи эйлеров маршрут в схеме, представленной на рис. 7-9, а. Маршрут ищется между узлами a

1. В качестве простой цепи, соединяющей узлы a и b , выбираем маршрут, включающий контакты x_1 и x_2 . Помечаем их цифрой 0 (в кружке).

2. Так как в схеме имеются непомеченные контакты, то среди них выбираем контакт x_4 , инцидентный узлу a . В подсхеме, порожденной непомеченными контактами, выявляем простой цикл, содержащий контакт $x_4, x_4 - x_3$.

Контакты этого цикла помечаем цифрой 1.

3. Так как в схеме все контакты помечены, то начинаем построение эйлерова маршрута. Построение маршрута начинается из узла a . Среди контактов,

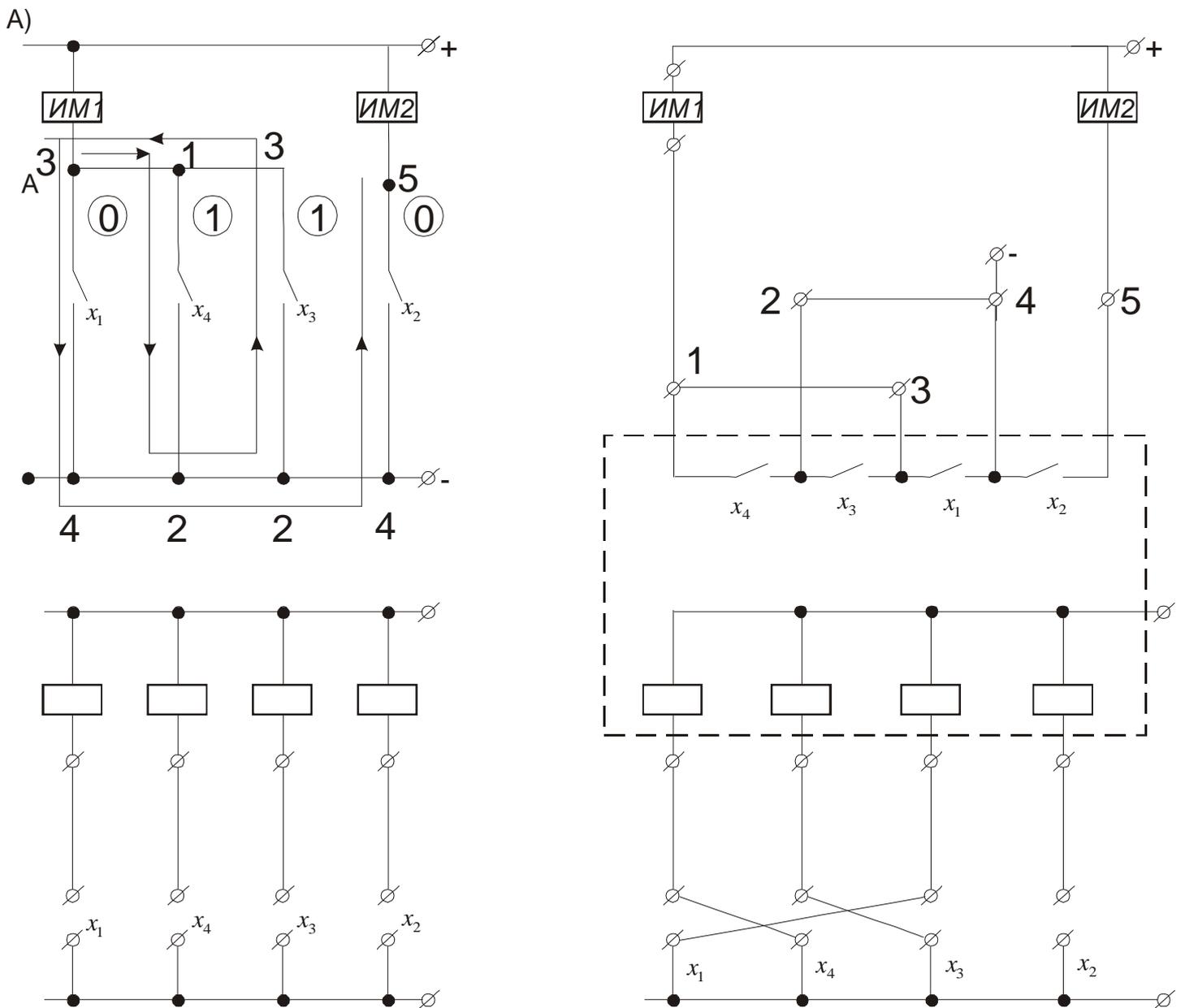


Рис. 7-9. Построение схемы в базисе настраиваемых модулей: *а* - реализуемая схема; *б* - модульная реализация

инцидентных этому узлу, выбираем контакт, помеченный наибольшей цифрой. Наибольшей цифрой помечены контакты x_4 и x_3 . Выбираем контакт x_4 . Для узла, инцидентного с этим контактом, наибольшую цифру имеет контакт x_3 , поэтому включаем его в маршрут. Продолжая этот процесс, получим маршрут: $x_4 - x_3 - x_1 - x_2$.

Продолжая рассмотрение свойств схем из ЭДП, отметим, что если схема содержит более двух нечетных узлов, то она не является уникурсальной. При числе нечетных узлов в схеме, большем двух ($g > 2$), эйлеров маршрут с помощью алгоритма Хоанг Туи определяется в расширенной схеме, получаемой из исходной путем введения $(g/2) - 1$ контактов, каждый из которых объединяет пару нечетных узлов. В результате образуется схема, содержащая лишь два нечетных узла, в которой обязательно существует эйлеров маршрут. Введенные контакты не нарушают условий работы исходной схемы, так как обмотки, соответствующие этим контактам, к источникам входных сигналов не подсоединяются.

На втором этапе реализации, двигаясь по эйлерову маршруту, считываем обозначения контактов в схеме и присваиваем их контактам цепочки слева направо в порядке считывания. При этом осуществляется также соответствующее подключение обмоток реле к ИИ ($x_1 - x_4$) (рис. 7-9, б).

Вновь двигаясь по эйлерову маршруту, нумеруем входной и выходной вывод каждого контакта схемы так, что проставляемые номера при движении от начала маршрута к его концу не уменьшаются. При этом выводу, соответствующему началу маршрута, присваивается номер 1; выходному выводу каждого контакта (считая по ходу маршрута) присваивается номер, на единицу больший по сравнению с номером его входного вывода; одинаковыми номерами обозначаются выходной вывод предыдущего и входной последующего контактов, расположенных на маршруте последовательно. На рис. 7-9, а выполнена нумерация выводов контактов для схемы, рассмотренной в предыдущем примере:

$$x_4(1;2), x_3(2;3), x_1(3;4), x_2(4;5).$$

Перед выполнением третьего этапа выписываем подмножества номеров выводов контактов реализуемой схемы, объединяемых между собой в соответствии с ее структурой. Для примера, рассмотренного выше, такими подмножествами являются: (1;3), (2;4), (5).

На третьем этапе выполняется собственно реализация схемы в базисе цепочки. Единая цепочка из $l = d$ контактов позволяет безызыбыточно (по числу контактов) реализовать произвольную (в том числе многополюсную и мостиковую) эйлерову схему из d контактов.

Единая цепочка из

$$l = d + \frac{g}{2} - 1 \quad (7-3)$$

контактов позволяет реализовать произвольную схему из d контактов, содержащую g нечетных узлов ($g > 2$).

Справедливость этих утверждений обеспечивается: а) рассмотренным выше свойством «проходимости» контактных схем, содержащих d контактов и g нечетных узлов, при введении дополнительных $(g/2) - 1$ контактов ($g \geq 2$); б) наличием всех промежуточных выводов у цепочки, позволяющих выполнять произвольные их объединения.

В ходе выполнения третьего этапа: а) слева направо нумеруются выводы цепочки, число контактов в которой l определяется из (7-3); б) объединяются с помощью перемычек те выводы цепочки, номера которых совпадают с номерами выводов контактов, найденными в ходе выполнения второго этапа.

Таким образом, в рассматриваемом примере схема будет реализована цепочкой из четырех контактов, в которой перемычки наложены между выводами 1 и 3, а также 2 и 4 (рис. 7-9, б).

Для класса схем из ЭДП справедливо следующее утверждение: если нечетные узлы в схеме отсутствуют, то суммарная длина перемычек, накладываемых на цепочку при реализации заданной схемы, минимальна в случае, когда обход схемы начинается из узла, который образован наибольшим числом контактов.

На рис. 7-10 приведен пример, демонстрирующий справедливость высказанного утверждения: при обходе из узла, образованного наибольшим числом контактов, суммарная длина перемычек

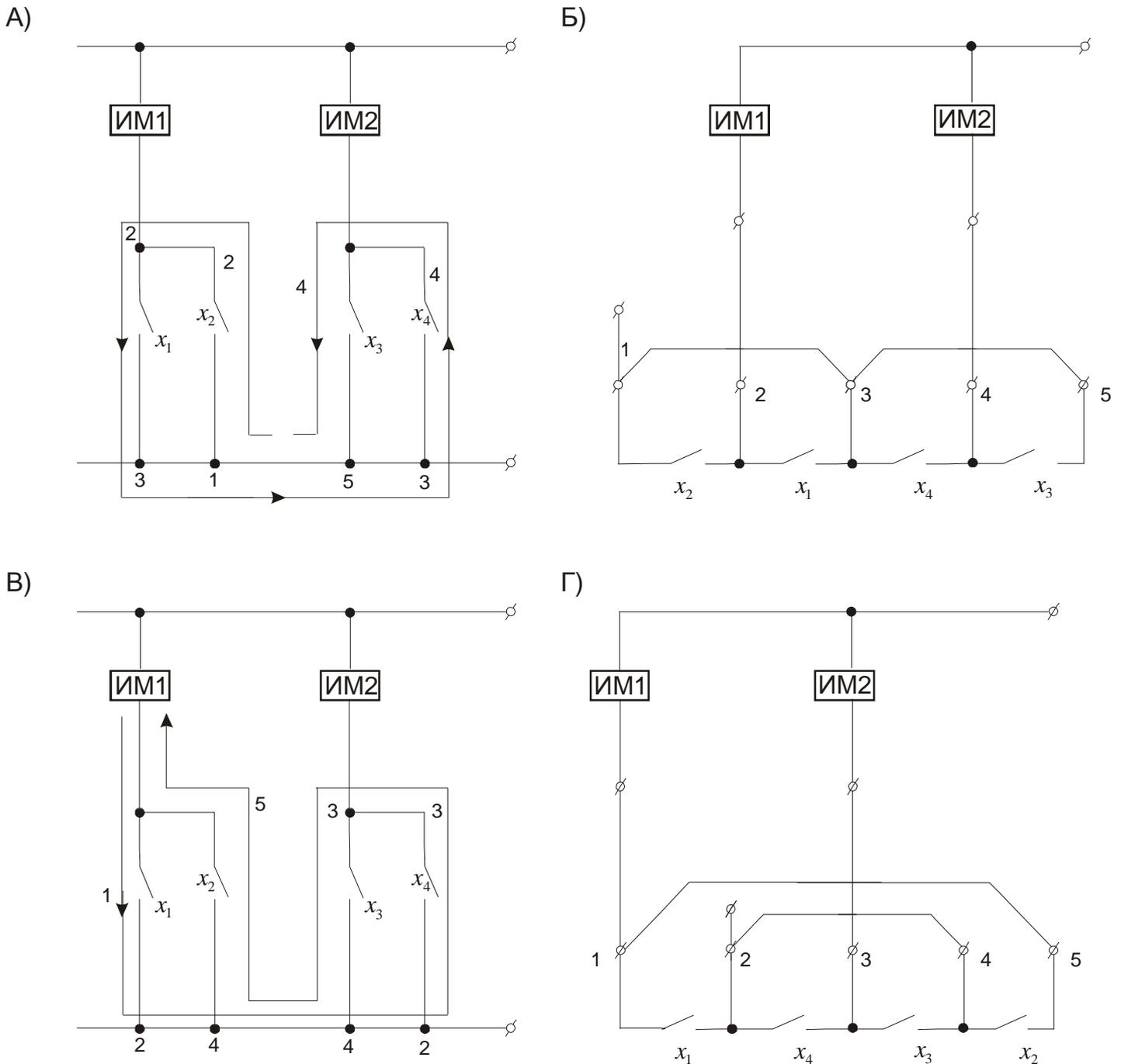


Рис. 7-10. Минимизация длины перемычек: а, б - реализация с перемычками минимальной длины; в, г - реализация с неминимальной длиной перемычек

равна четырем (рис. 7-10, а, б), а при обходе, начинающемся из любого другого узла, - шести (рис. 7-10, в, г).

Рассмотрим вопрос о реализации схем в базисе модулей из замыкающих и размыкающих контактов. Пусть построена произвольная контактная схема, содержащая замыкающие и размыкающие контакты, суммарное число которых равно d , число нечетных узлов в схеме равно g , а реализация выполняется на модулях из переключающих контактов.

Предположим, что имеется неограниченное число модулей, содержащих одно, два, три реле, и требуется реализовать схему с минимально возможной избыточностью. Процедура реализации состоит из трех основных этапов.

Первый этап. Выявляются подсхемы, на которые может быть разделена схема для обеспечения реализации с минимально возможной избыточностью на основе процедуры, изложенной выше.

Число этих подсхем

$$L = g / 2. \quad (7-4)$$

При этом каждой из найденных подсхем, содержащей l_i контактов, соответствует модуль, состоящий из l_i реле.

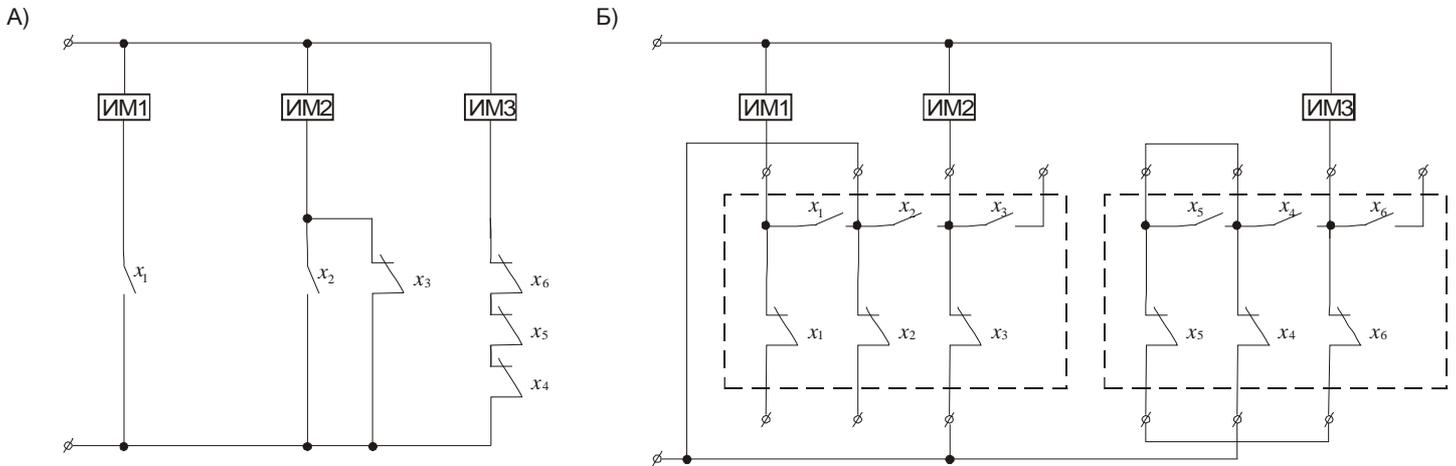


Рис. 7-11. Построение схемы в базе настраиваемых модулей из переключающих контактов: а – реализуемая схема; б - модульная реализация

Второй этап. Каждый из полученных модулей последовательно разбивается на совокупность несвязанных модулей из трех, двух и одного реле. При этом предпочтение отдается модулям, содержащим наибольшее число реле.

Третий этап. Каждый из модулей настраивается на реализацию соответствующей подсхемы путем использования таблицы настроек (см. § 7-2). Возможность такой настройки гарантирована в связи с универсальностью этих модулей в классе схем, содержащих замыкающие и размыкающие контакты при $h = 1; 2; 3$. На рис. 7-11 приведен пример реализации схемы в базе модулей из замыкающих и размыкающих контактов на основе изложенной процедуры.

Оценки сложности реализации произвольной двухполюсной схемы в базе одного модуля. Выше было показано, что произвольная (в том числе многополюсная) схема из d контактов, содержащая g нечетных узлов, может быть реализована единой цепочкой, длина которой при $g \geq 2$ определяется по формуле

$$l = d + \frac{g}{2} - 1. \quad (7-5)$$

Для нахождения искомой оценки определим соотношение между числом нечетных узлов и числом контактов в произвольной двухполюсной схеме. Для этого авторами рассматривались схемы, содержащие максимальное число узлов нечетной степени при

Таблица 7-1

**Максимальные значения числа нечетных узлов
в произвольной двухполюсной схеме**

d	1-3	4-6	7-9	10-12	13-15	16-18	19-21	22-24
g_{\max}	2	4	6	8	10	12	14	16

фиксированном числе контактов. В табл. 7-1 указаны максимальные значения числа нечетных узлов в этих схемах для значений $d = 1 \div 24$.

Среди этих схем доля нечетных узлов, приходящихся на один контакт (отношение g/d), максимальна при $d = 1 + 3t$, где $t = 0, 1, 2, \dots$

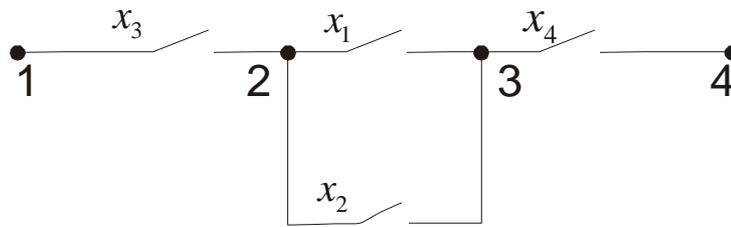
Схемы, состоящие из $d = 1 + 3t$ контактов, образуют класс схем, содержащих минимальное число контактов при фиксированном числе узлов нечетной степени. Схемы этого класса характеризуются рядом свойств.

В двухполюсных схемах, содержащих минимальное число контактов при фиксированном числе узлов нечетной степени ($d \geq 4$), все узлы, за исключением двух, степень которых равна 1, имеют степень, равную трем. Действительно, для того чтобы образовать нечетный узел, минимально необходимое число контактов должно быть равно одному. Однако в двухполюсной схеме не может быть более двух узлов степени 1, и поэтому в схеме, обладающей указанным свойством при $d \geq 4$, все остальные узлы нечетной степени должны быть образованы тремя контактами.

На основе исследования этих схем ниже определяется верхняя оценка сложности реализации произвольной двухполюсной схемы в базисе одного модуля. Эти схемы являются наихудшими в смысле их «проходимости» единой цепочкой, так как «проходимость» схемы определяется узлами нечетной степени.

Параллельно-последовательные схемы рассматриваемого класса при $d \geq 4$ могут быть получены путем итеративной подстановки схемы, представленной на рис. 7-12, а, вместо отдельного контакта этой же схемы. В результате одной такой подстановки

А)



Б)

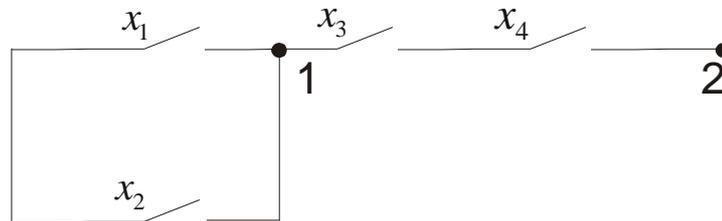


Рис. 7-12. Реализация формулы $y = (x_1 \vee x_2)x_3x_4$: а - схема с четырьмя нечетными узлами; б - схема с двумя узлами

число узлов нечетной степени в схеме увеличивается на два, а число контактов - на три.

Для рассматриваемого класса схем справедливо соотношение

$$g'_{\max} = \frac{2}{3}(d + 2). \quad (7-6)$$

Действительно, из теории графов известно, что

$$\sum_{i=1}^{g'_{\max}} r_i = 2d, \quad (7-7)$$

где r_i - степень i -го узла.

Это выражение может быть преобразовано исходя из утверждения, приведенного выше, следующим образом:

$$3(g'_{\max} - 2) + 1 \times 2 = 2d; \quad (7-8)$$

$$g'_{\max} = \frac{2}{3}(d + 2). \quad (7-8)$$

Для схем, содержащих максимальное число узлов нечетной степени при фиксированном числе контактов,

$$g''_{\max} = \frac{2}{3}(d + 1) \quad \text{при} \quad d = 2 + 3t; \quad (7-10)$$

$$g''_{\max} = \frac{2}{3}d \quad \text{при} \quad d = 3 + 3t; \quad (7-11)$$

Приведенные соотношения могут быть объединены в одно

$$g^c_{\max} = \left\langle \frac{2}{3}(d+2) \right\rangle, \quad (7-12)$$

где $\langle \rangle$ - знак округления до ближайшего четного в меньшую сторону.

Таким образом, справедливо утверждение: число узлов нечетной степени в произвольной двухполюсной схеме из d контактов удовлетворяет выражению

$$g \leq \left\langle \frac{2}{3}(d+2) \right\rangle, \quad (7-13)$$

Максимальная длина единой цепочки, необходимой для реализации произвольной двухполюсной схемы из d контактов, удовлетворяет выражению

$$l_{\max} = \left[\frac{4}{3}d - \frac{1}{3} \right]. \quad (7-14)$$

Действительно, подставляя (7-12) в выражение (7-3), получим

$$l_{\max} = d + \frac{\left\langle \frac{2}{3}(d+2) \right\rangle}{3} - 1 = \left[\frac{4}{3}d - \frac{1}{3} \right]. \quad (7-15)$$

В табл. 7-2 приведены значения l_{\max} при значениях $d = 1 \div 10$.

Таблица 7-2

Верхние оценки длины единой цепочки, реализующей произвольную двухполюсную схему из d контактов

d	1	2	3	4	5	6	7	8	9	10
l_{\max}	1	2	3	5	6	7	9	10	11	13

Длина единой цепочки, требующейся для реализации произвольной двухполюсной схемы из d контактов, удовлетворяет выражению

$$d \leq l^c \leq \left[\frac{4}{3}d - \frac{1}{3} \right]. \quad (7-16)$$

Так как для реализации схемы из d контактов требуется d контактов «россыпью», то справедливо утверждение: максимальная избыточность при использовании единой цепочки для реализации двухполюсной схемы удовлетворяет выражению

$$\varphi_1 \leq \frac{4}{3}. \quad (7-17)$$

Оценка сложности реализации произвольной булевой функции, заданной формулой в базисе $\{\&, \vee, \bar{}\}$, Ввиду того что для каждой булевой формулы из h букв может быть построена соответствующая двухполюсная параллельно-последовательная схема из $d = h$ контактов, искомая оценка может быть получена из выражения (7-16) при замене d на A .

Длина единой цепочки, требующейся для реализации произвольной булевой формулы из h букв, удовлетворяет выражению

$$h \leq l^{\Phi} \leq \left[\frac{4}{3}h - \frac{1}{3} \right]. \quad (7-18)$$

Если булева функция n переменных задана нормальной формулой из h букв ($n \leq h$), то нижняя оценка реализации булевой функции равна n , так как число контактов в цепочке может быть равно числу независимых переменных. Таким образом, длина единой цепочки, требующейся для реализации булевой функции n переменных, заданной нормальной формулой в базисе $\{\&, \vee, \bar{}\}$ из h букв, удовлетворяет выражению

$$n \leq l^{\text{функ}} \leq \left[\frac{4}{3}h - \frac{1}{3} \right]. \quad (7-19)$$

Полученная верхняя оценка, по-видимому, является завышенной, так как каждая формула, возможно, может быть реализована схемой, число нечетных узлов g^{Φ} в которой удовлетворяет неравенству

$$g^{\Phi} < g_{\max}^c. \quad (7-20)$$

Например, формула $y = (x_1 \vee x_2)x_3x_4$ может быть реализован двумя схемами (рис. 7-12, а, б) первая из которых имеет при четырех контактах максимальное число нечетных узлов - четыре, а вторая - лишь два.

На основе рассмотрения схем, реализующих формулы при малых значениях h ($h = 1 \div 8$), было сделано предположение, что

$$g_{\max}^{\Phi}(h) = g_{\max}^c(h) - 2. \quad (7-21)$$

Если это предположение справедливо для произвольных h , то полученная верхняя оценка может быть снижена:

$$l_{\max}^{\Phi} = h + \frac{g_{\max}^{\Phi}}{2} - 1 = h + \frac{\left\langle \frac{2(h+2)}{3} \right\rangle - 2}{2} - 1 = \left[\frac{4}{3}(h-1) \right]. \quad (7-22)$$

Таким образом, возможно, что справедливо выражение

$$h \leq l^{\Phi} \leq \left[\frac{4}{3}(h-1) \right]. \quad (7-23)$$

Оценки сложности реализации монотонных функций. В работе [23] Н. А. Карпова предложила каталог минимальных схем из замыкающих контактов для всех 210 типов монотонных функций пяти и менее переменных. При этом она установила, что любая монотонная функция n переменных требует для своей реализации не более $l(n)$ контактов «россыпью» (табл. 7-3).

Таблица 7-3

Сложность реализации монотонных функций

n	1	2	3	4	5
$l(n)$	1	2	5	8	13

Авторами исследовался вопрос о реализации монотонных функций пяти и менее переменных в базисе единой цепочки на основе каталога минимальных схем. При этом длина единой цепочки, достаточная для реализации произвольной монотонной функции, существенно зависящей от n переменных, может быть оценена на основе выражения

$$l_{\max}^{\text{exp}}(n) = \left[\frac{4}{3}l(n) - \frac{1}{3} \right]. \quad (7-24)$$

В табл. 7-4 приведены значения $l_{\max}^{\text{exp}}(n)$ для $n = 1 \div 5$.

Ввиду того что реальная длина единой цепочки $l_{\max}(n)$, достаточная для реализации любой монотонной функции, существенно зависящие от n переменных, удовлетворяет выражению

Таблица 7-4

n	1	2	3	4	5
$l_{\max}^{\text{exp}}(n)$	1	2	6	10	17

$$l_{\max}(n) \leq l_{\max}^{\text{exp}}(n), (7-25)$$

с целью определения точных значений $l_{\max}(n)$ была выполнена реализация представителей каждого типа монотонных функций из пяти и менее переменных в базе единой цепочки.

В табл. 7-5 указано число типов монотонных функций $S_m(n)$, существенно зависящих от n переменных.

Таблица 7-5

Число типов монотонных функций

n	0	1	2	3	4	5
$S_m(n)$	2	1	2	5	20	180

В табл. 7-6 ÷ 7-11 приведены результаты выполненного исследования. При этом использованы следующие обозначения: $N(l)$ -число типов монотонных функций, оптимально реализуемых l контактами «россыпью»; $M(l)$ -число типов монотонных функций, оптимально реализуемых цепочкой из l контактов.

Таблица 7-6

$n = 0$

l	0
$N(l)$	2
$M(l)$	2

Таблица 7-7

$n = 1$

l	1
$N(l)$	1
$M(l)$	1

Таблица 7-8

$n = 2$

l	2
$N(l)$	2
$M(l)$	2

Таблица 7-9

$n = 3$

l	3	4	5
$N(l)$	4	-	1
$M(l)$	4	-	1

$n = 4$					
l	4	5	6	7	8
$N(l)$	10	2	6	-	2
$M(l)$	10	2	6	-	2

На основе приведенных результатов были установлены точные значения $l_{\max}(n)$ (табл. 7-12).

В ходе процесса реализации монотонных функций в базисе цепочки установлена справедливость следующих утверждений.

Цепочка длиной $l_{\max}(n)$ является НЛМ, универсальным для класса монотонных функций n и менее переменных. Максимальная

Таблица 7-11

$n = 5$											
l	5	6	7	8	9	10	Π	12	13	14	15
$N(l)$	25	16	52	32	37	11	2	4	1	-	-
$M(l)$	25	15	52	28	37	12	5	4	1	-	1

Таблица 7-12

Сложность реализации монотонных функций в базисе единой цепочки

n	0	1	2	3	4	5
$l_{\max}(n)$	0	1	2	5	8	15

Таблица 7-13

Монотонные функции, реализуемые с избыточностью в базисах цепочки и контактов «россыпью»

N	2	10	58	72	92	99	100	110	113	114	133	137	138	173	182	204
$l_{\text{рос}}$	13	11	8	9	8	10	10	9	8	8	8	7	9	6	9	9
$l_{\text{ц}}$	15	13	9	11	9	11	11	10	9	9	9	9	10	7	10	11

ная длина цепочек $l_{\max}(n)$ при $n = 3; 4; 5$ достигается при реализации мажоритарных функций «2 из 3», «2 из 4», «3 из 4» и «3 из 5» соответственно. При этом функция «2 из 4» должна быть реализована

Π-схемой, описываемой формулой $y = x_1 x_2 \vee (x_1 \vee x_2)(x_3 \vee x_4) \vee x_3 x_4$.

Реализация монотонных функций при $n \leq 4$ в базисе единой цепочки не приводит к избыточности по сравнению с реализацией в базисе контактов «россыпью». Из 180 PN -типов монотонных функций, существенно зависящих от пяти переменных, лишь 16 реализуются с избыточностью.

В табл. 7-13 приведены номера соответствующих этим функциям минимальных схем по каталогу, а также число контактов, требующихся для их реализации в базисах контактов «россыпью» и цепочки.

Оценки сложности реализации произвольных булевых функций.

Таблица 7-14

Верхние оценки сложности реализации произвольной функции n переменных

n	1	2	3	4	5
$\lambda(n)$	1	4	8	13	28

В работах К. Шеннона [57], Ю. Л. Васильева [10], Г. Н. Поварова [41] было установлено, что произвольная булева функция n переменных требует для своей реализации не более $\lambda(n)$ контактов

«россыпью» (табл. 7-14). На основе верхней оценки длины цепочки, требующейся для реализации схемы из $\lambda(n)$ контактов, предполагая, что прямые и инверсные значения *III* равнодоступны, можно утверждать, что

$$l_{\max}(n) \leq l_{\max}^{exp}(n) = \left[\frac{4}{3} \lambda(n) - \frac{1}{3} \right]. \quad (7-26)$$

Значения l_{\max}^{exp} приведены в табл. 7-15.

Определим точные значения для $l_{\max}(2)$ и $l_{\max}(3)$.

Среди функций двух переменных только функции $y_1 = \bar{x}_1 \bar{x}_2 \vee x_1 x_2$ и $y_2 = \bar{x}_1 x_2 \vee x_1 \bar{x}_2$ требуют для своей реализации четыре контакта, а остальные - лишь два. Ввиду того что функции y_1 и y_2 могут быть реализованы схемами из четырех контактов без нечетных узлов, $l_{\max}(2) = 4$.

Таблица 7-15

Сложность реализации произвольных функций n переменных в базисе единой цепочки (верхние оценки)

n	1	2	3	4	5
$l_{\max}^{exp}(n)$	1	5	10	17	37

Среди 16 *PN*-типов функций, существенно зависящих от трех переменных, только три представителя типов функций могут потребовать для своей реализации в базисе цепочек более восьми контактов:

$$z_1 = x_1(x_2 x_3 \vee \bar{x}_2 \bar{x}_3) \vee \bar{x}_1 x_2 \bar{x}_3; z_2 = x_1(x_2 \vee x_3) \vee x_2 x_3 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3; z_3 = x_1(x_2 x_3 \vee \bar{x}_2 \bar{x}_3) \vee \bar{x}_1(x_2 \bar{x}_3 \vee \bar{x}_2 x_3).$$

Остальные 13 типов функций представимы булевыми формулами из шести и менее букв и поэтому не могут потребовать для своей реализации цепочку длиной, большей семи, так как при шести контактах в схеме не может быть более четырех нечетных узлов.

Функции z_1 и z_2 могут быть реализованы параллельно-последовательными схемами из восьми контактов, содержащими по два нечетных узла, поэтому для реализации этих функций достаточно иметь цепочку из восьми контактов. Функция z_3 может быть реализована мостиковой схемой из восьми контактов, содержащей четыре нечетных узла (рис. 7-13), для «покрытия» которой требуется цепочка из девяти контактов. Следовательно, $l_{\max}(3) = 9$.

Определение точных значений $l_{\max}(4)$ и $l_{\max}(5)$ не выполнялось, так как каталоги оптимальных реализаций для функций четырех и пяти переменных не были опубликованы.

Значения $l_{\max}(n)$ сведены в табл. 7-16.

Таким образом, можно утверждать, что цепочка длиной $l_{\max}(n)$ является НЛМ, универсальным для класса произвольных функций n переменных.

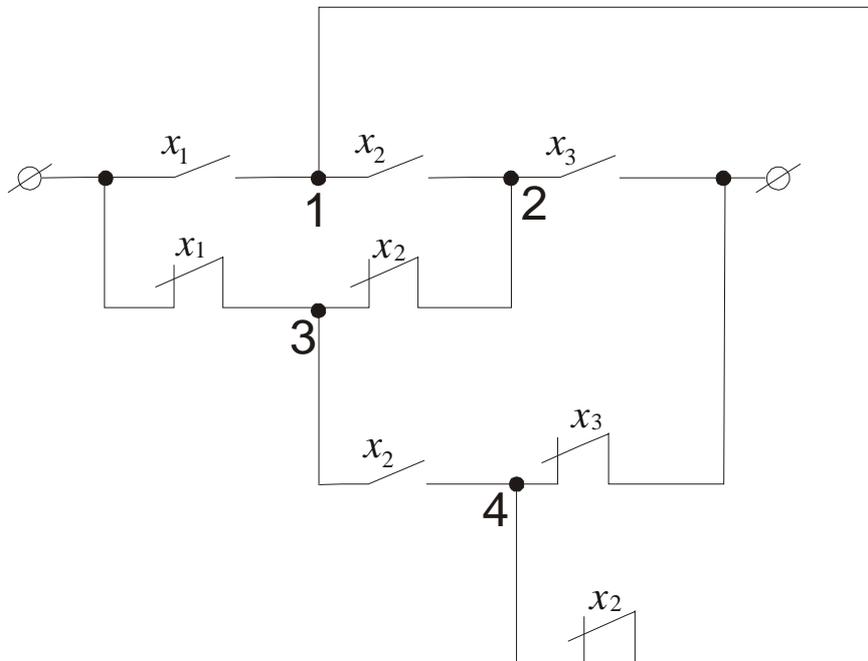


Рис. 7-13. Реализация функции, обладающей наиболее сложной модульной схемой из всех функций трех переменных

Оценка сложности реализации произвольной многополюсной схемы в базисе одного модуля.

Для получения искомой оценки авторами рассматривались многополюсные схемы, содержащие максимальное число нечетных узлов при фиксированном числе контактов. Среди этих схем для получения верхней оценки наибольший интерес представляют схемы, содержащие $d = 1 + 2t$ контактов

Таблица 7-16

Сложность реализации произвольных функций я переменных в базисе единой цепочки

n	1	2	3	4	5
$l_{\max}(n)$	1	4	9	17	37

($t = 0, 1, 2, \dots$). Они имеют минимальное число контактов при фиксированном числе нечетных узлов. Для схем этого класса выполняется соотношение

$$g'_{\max} = d + 1, \quad (7-27)$$

в то время как для схем, содержащих четное число контактов $d = 2t$, выполняется соотношение

$$g''_{\max} = d. \quad (7-28)$$

Таким образом, в общем случае справедливо равенство

$$g_{\max} = \langle d + 1 \rangle. \quad (7-29)$$

Подставляя найденное равенство в выражение (7-3), получим

$$l_{\max} = d + \frac{\langle d + 1 \rangle}{2} - 1 = \left\lceil \frac{3d - 2}{2} \right\rceil. \quad (7-30)$$

Таким образом, длина единой цепочки, требующейся для реализации произвольной многополюсной схемы из d контактов, удовлетворяет выражению

$$d \leq l \leq \left\lceil \frac{3d - 2}{2} \right\rceil. \quad (7-31)$$

Так как для реализации схемы из d контактов с помощью контактов «россыпью» требуется d контактов, то максимальная избыточность при использовании единой цепочки для реализации многополюсной схемы удовлетворяет выражению

$$d \leq l \leq \left\lceil \frac{3d-2}{2} \right\rceil. \quad (7-32)$$

Оценка сложности реализации системы булевых формул в базисе одного модуля. Пусть требуется реализовать многополюсную контактную схему с общей шиной систему из N нормальных булевых формул, суммарное число букв в которой равно

$H \left(H = \sum_{i=1}^N h_i \right)$. Так как в худшем случае каждая формула системы может быть реализована независимо

параллельно-последовательной схемой, состоящей из контактов, то искомая верхняя оценка может быть получена путем замены d на H в правой части выражения (7-31). Длина единой цепочки, требующейся для реализации системы из N нормальных булевых формул, суммарное число букв в которой равно H , удовлетворяет выражению

$$H \leq l \leq \left\lceil \frac{3H-2}{2} \right\rceil. \quad (7-33)$$

Оценка сложности реализации схемы в базисе набора однотипных модулей. Процедура реализации схемы в базисе цепочек длиной l состоит из двух основных этапов: а) реализации схемы в базисе единой цепочки; б) «покрытия» единой цепочки набором цепочек, каждая из которых имеет длину l .

Из сказанного следует, что оценки сложности в этом случае могут быть получены на основе оценок, найденных для случая использования единой цепочки, путем их деления на l и округления до ближайшего целого в большую сторону.

1. *Оценка сложности реализации произвольной двухполюсной схемы из d контактов*

$$\left\lceil \frac{d}{l} \right\rceil \leq L_2 \leq \left\lceil \frac{\left\lceil \frac{4}{3}d - \frac{1}{3} \right\rceil}{l} \right\rceil. \quad (7-34)$$

2. *Оценка сложности реализации произвольной булевой функции n переменных, заданной формулой в базисе $\{\&, \vee, \bar{}\}$ из h букв:*

$$\left\lceil \frac{n}{l} \right\rceil \leq L_2 \leq \left\lceil \frac{\left\lceil \frac{4}{3}h - \frac{1}{3} \right\rceil}{l} \right\rceil. \quad (7-35)$$

Есть основания предполагать, что верхняя оценка может быть снижена:

$$\left\lceil \frac{n}{l} \right\rceil \leq L_2 \leq \left\lceil \frac{\left\lceil \frac{4}{3}(h-1) \right\rceil}{l} \right\rceil. \quad (7-36)$$

Сопоставим полученные оценки с соответствующими оценками для модулей из ФЭ. Предположим, что используется модуль из ФЭ, универсальный для класса формул в базисе $\{\&, \vee, \bar{}\}$ из l букв. При этом справедливо соотношение

$$\left\lceil \frac{h-1}{K-1} \right\rceil \leq L \leq \left\lceil \frac{2(h-1)}{l} \right\rceil. \quad (7-37)$$

Из сопоставления оценок (7-35) и (7-37) следует, что верхняя оценка числа модулей из ЭДП, требующихся для реализации произвольной формулы в базисе $\{\&, \vee, \bar{}\}$ из h букв, приблизительно в

1,5 раза меньше соответствующей оценки для модулей из ФЭ, обладающих теми же функциональными возможностями для рассматриваемого класса формул, при практически одинаковых нижних оценках. Одинаковые функциональные возможности модулей из ЭДП и ФЭ для класса формул из l букв в базе $\{\&, \vee, \bar{\quad}\}$ следует понимать в том смысле, что если, например, имеется стандартный корпус с 14 внешними выводами, то в таком корпусе может быть размещен либо один модуль из ФЭ, либо один модуль из ЭДП, каждый из которых универсален для класса формул из шести букв в базе $\{\&, \vee, \bar{\quad}\}$.

Из сказанного можно сделать вывод, что модули из ЭДП в среднем значительно более эффективны по сравнению с модулями из ФЭ.

3. Оценка сложности реализации произвольной многополюсной схемы из d контактов

$$\left\lceil \frac{d}{l} \right\rceil \leq L_3 \leq \left\lceil \frac{\left\lceil \frac{3d-2}{2} \right\rceil}{l} \right\rceil = \left\lceil \frac{3d-2}{2l} \right\rceil. \quad (7-38)$$

4. Оценка сложности реализации системы N булевых формул, суммарное число букв в которой равно H , а число независимых переменных n :

$$\left\lceil \frac{n}{l} \right\rceil \leq L_4 \leq \left\lceil \frac{3H-2}{2l} \right\rceil. \quad (7-39)$$

Оценки сложности реализации в базе минимального числа модулей, номенклатура которых определяется в ходе процесса построения. Если связная контактная схема имеет g нечетных узлов, где $g \geq 2$, то она может быть безызбыточно реализована с помощью $g/2$ цепочек, каждая из которых соединяет по два узла нечетной степени, т. е.

$$L = g/2. \quad (7-40)$$

Если связная контактная схема имеет g узлов нечетной степени, где $g \geq 2$, то она может быть безызбыточно реализована с помощью по крайней мере двух различных наборов цепочек, каждый из которых содержит по $g/2$ цепочек.

На основе (7-40) и оценок числа нечетных узлов, найденных выше, могут быть получены следующие оценки числа цепочек.

1. Оценка сложности реализации произвольной двухполюсной схемы из d контактов

$$1 \leq L' \leq \frac{\left\langle \frac{2}{3}(d+2) \right\rangle}{2} = \left\lceil \frac{d+2}{3} \right\rceil. \quad (7-41)$$

2. Оценка сложности реализации произвольной булевой формулы из h букв

$$1 \leq L'' \leq \left\lceil \frac{h+2}{3} \right\rceil. \quad (7-42)$$

Есть основания полагать, что верхняя оценка в этом случае может быть снижена:

$$1 \leq L'' \leq \frac{\left\langle \frac{2}{3}(h+2) \right\rangle - 2}{2} = \left\lceil \frac{h+2}{3} \right\rceil - 1 = \left\lceil \frac{h-1}{3} \right\rceil. \quad (7-43)$$

3. Оценка сложности реализации произвольной многополюсной схемы из d контактов

$$1 \leq L''' \leq \frac{\langle d+1 \rangle}{2} = \left\lceil \frac{d}{2} \right\rceil. \quad (7-44)$$

4. Оценка сложности реализации системы булевых формул

$$1 \leq L''' \leq \left\lceil \frac{H}{2} \right\rceil. \quad (7-45)$$

где H - суммарное число букв в формулах системы.

Оценка сложности реализации произвольной схемы, содержащей g нечетных узлов, при использовании набора цепочек и трехлучевых звезд

$$L \leq g / 2 \quad (7-46)$$

Если применение одной звезды (рис. 7-14) позволяет уменьшить число нечетных узлов в остаточной схеме на четыре, то общее число дискретных компонентов, требующихся для реализации схемы, уменьшается на одну: $L = \frac{g-4}{2} + 1 = \frac{g}{2} - 1$. Если применение одной звезды позволяет уменьшить число нечетных узлов на два, то общее число дискретных компонентов не изменяется:

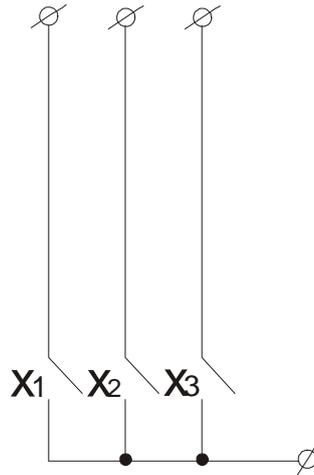


Рис. 7-14. Трехлучевая звезда

Если применение одной звезды не позволяет уменьшить число нечетных узлов, а уменьшает лишь степени некоторых из них, то расширение базиса нецелесообразно. «Покрытие» в этом случае должно выполняться только цепочками:

$$L = g / 2. \quad (7-47)$$

На рис. 7-15 приведена одна из структур типа «звезда из цепочек», являющаяся обобщением структур типа «звезда». Структуры этой разновидности могут содержать различное число цепочек, причем эти цепочки могут обладать различной длиной ($l = 1 \div 5$). При этом любая формула из h букв ($h \leq 5$) может быть безызбыточно реализована цепочкой, в которой один из крайних выводов соединен с общей шиной.

При $h = 6$ существует по крайней мере одна формула $y = x_1(x_2 \vee x_3) \vee x_4(x_5 \vee x_6)$, которая не может быть реализована безызбыточной цепочкой из шести контактов, входящей в состав многолучевой звезды. Эта формула может быть реализована цепочкой из семи контактов, в которой контакт, соединенный с общей шиной, не используется.

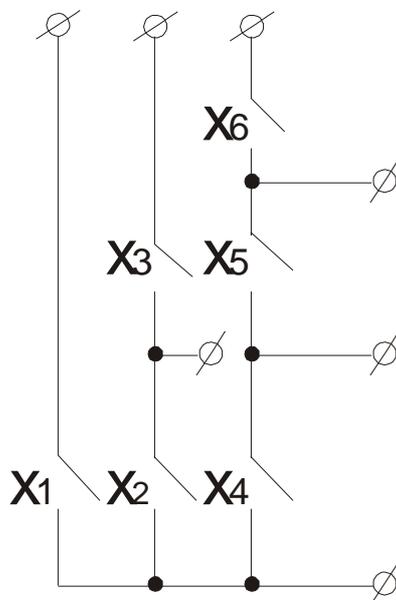


Рис. 7-15. Звезда из цепочек

В заключение отметим, что для модулей тип «цепочка», «звезда» и «звезда из цепочек» справедливо соотношение

$$M = N + d, \quad (7-48)$$

где M - число выводов в модулях; N - число модулей; d - суммарное число контактов в модулях.

Из этого соотношения следует, что при $d = const$ число внешних выводов определяется числом модулей, в которые объединены контакты. При $d = const$ и $N = const$ имеем $M = const$ вне зависимости от номенклатуры модулей.

7.4. УНИФИКАЦИЯ БЛОКОВ ЛОГИКИ ИЗ ЭЛЕМЕНТОВ С ДВУСТОРОННЕЙ ПРОВОДИМОСТЬЮ

Сокращение ресурсов, затрачиваемых на этапах проектирования, изготовления и эксплуатации систем логического управления, невозможно без проведения схемно-конструктивной унификации.

В работе [54] были рассмотрены методы построения унифицированных единиц замены и отмечалась целесообразность создания двух классов модулей, которые условно названы специализированными и универсальными. Специализированные модули характеризуются определенным целевым назначением и создаются в том случае, если в системах регулярно повторяются какие-либо устройства или алгоритмы. Однако в системах логического управления наряду с «устойчивостью» принципов управления, контроля и сигнализации, приводящей к унификации и специализации соответствующих модулей, имеет место большое разнообразие в объектах управления и алгоритмах их работы.

Указанное разнообразие приводит к различиям в логических схемах, которые в системах рассматриваемого класса, как отмечалось в § 1-1, почти повсеместно являются комбинационными.

Нерегулярные комбинационные схемы, занимающие в системах лишь 15-20% объема, определяют при отсутствии унификации 50-60% номенклатуры кассет или субблоков, и поэтому одним из важнейших вопросов в проблеме унификации систем этого класса является разработка методов построения

универсальных единиц замены, предназначенных для реализации нерегулярных комбинационных схем.

Пусть имеется схемно-конструктивная единица (кассета), обладающая определенной схемной вместимостью, выраженной в числе базисных элементов и контактов разъемов. Требуется выбрать и «вписать» в определенном соотношении некоторые схемы таким образом, чтобы кассета обладала широкими функциональными возможностями и малыми избыточностью и внешним монтажом при выполнении требований по надежности, простоте контроля и диагностики.

В § 5-3 были рассмотрены вопросы унификации нерегулярных комбинационных схем из ФЭ и показано, что решение указанной задачи возможно лишь при применении принципа многофункцио-

нальности. Этот методологический принцип был использован при разработке унифицированных единиц замены на интегральных схемах и позволил удовлетворить указанным выше требованиям и ограничениям. В настоящем параграфе делается попытка использовать принцип многофункциональности при унификации схем из ЭДП, и в частности из релейно-контактных элементов.

Одним из возможных путей унификации схем из релейно-контактных элементов является функциональный подход, при котором на контактах строят ячейки, реализующие наиболее часто встречающиеся в схемах элементарные булевы функции: «И», «ИЛИ», «Запрет» и т. д. Однако такой подход не обеспечивает решения проблемы унификации типовых единиц замены даже для схем из ФЭ, для которых он кажется естественным.

Реализация на контактах порождающих функций многофункциональных модулей, универсальных для класса формул, как показано в § 7-1, также не является целесообразной ввиду избыточности, неприемлемой для практики. При этом необходимо отметить, что такой подход решал рассматриваемую проблему для схем из ФЭ.

В настоящее время на практике при унификации логических схем из ЭДП применяется метод формирования кассет, обеспечивающий их универсальность путем вывода практически всех внешних выводов элементов на контакты разъемов. Применение этого подхода объясняется тем, что реализация схем в базе таких кассет чрезвычайно проста и состоит в соединении отдельных элементов на разъемах приборов в соответствии с заданной формулой или схемой. Однако использование таких унифицированных единиц, содержащих элементы почти «россыпью», приводит к тому, что практически весь монтаж не унифицирован и оказывается внутри приборов на разъемах, что вызывает значительные трудности как при монтаже и настройке, так и при поиске неисправностей в процессе эксплуатации. Кроме того, применение кассет, сформированных таким образом, приводит к избыточности, которая существует, несмотря на то, что в модулях с такой структурой гарантирована возможность использования практически каждого элемента. Это объясняется тем, что при таком подходе обычно не удается согласовать вместимость кассеты по числу элементов с числом контактов разъемов, и поэтому при использовании малогабаритных реле ограничение по числу внешних выводов достигается при относительно пустой кассете.

Для решения указанных противоречий предлагается использовать в составе кассет НЛМ из ЭДП, рассмотренные в § 7-2. Их применение в составе унифицированных кассет, используемых при построении систем логического управления, должно базироваться на учете: а) свойств применяемых в настоящее время элементов; б) требований, предъявляемых к структуре логических систем рассматриваемого класса, по алгоритмической гибкости, ремонтпригодности, надежности и т. д.

Среди технических характеристик электромагнитных реле, используемых в настоящее время, основным ограничивающим фактором для применения структур, рассмотренных в § 7-2, является то, что для большинства реле недопустимо, чтобы их обмотки долгое время находились под током. Учитывая это ограничение, цепочку из замыкающих контактов (см. рис. 7-3) можно использовать в качестве универсального модуля лишь для класса положительно-монотонных формул или при реализации схем из замыкающих контактов. Для отрицательно-монотонных формул и схем из размыкающих контактов может использоваться цепочка из размыкающих контактов.

Учет указанного ограничения не позволяет использовать в настоящее время структуру, представленную на рис. 7-4, и поэтому для реализации немонотонных формул должна применяться схема из замыкающих и размыкающих контактов, в которой в результате настройки ни один из элементов не находится долгое время под током (рис. 7-5).

Вторая особенность, влияющая на использование полученных результатов на практике, связана с уровнем интеграции применяемой элементной базы. Вопросы повышения уровня интеграции релейно-контактных элементов решены промышленностью далеко не достаточно. Если для интегральных микросхем очевидна тенденция объединять элементы внутри корпуса, то для релейно-контактных элементов этот подход практически не используется, что приводит к большому объему неупорядоченного внешнего монтажа и снижению надежности систем, построенных на их основе. Единственный путь повышения уровня интеграции этих элементов, применяемый промышленностью, состоит в выпуске многоконтактных реле, для эффективного использования которых были разработаны специализированные методы синтеза [41,57]. Однако этот подход не решает указанную проблему полностью, так как контакты и одного и различных реле не объединяются между собой внутри корпуса. Формирование цепочек является новым методом повышения уровня интеграции релейно-контактных элементов.

В § 7-3 мы попытались изложить и другую важнейшую сторону проблемы: как из этих модулей строить схемы? Особенность предложенных подходов состоит в том, что они наиболее эффективны при применении одноконтактных реле с одним замыкающим контактом (например, реле типа РЭС-60) или с одним переключающим контактом (например, реле типа РЭС-49).

Рассмотрим вопрос о дальнейшем повышении уровня интеграции схемных решений - использовании двухконтактных реле при построении настраиваемых модулей. Такие реле представляют в настоящее время наибольший интерес для проектантов многих систем логического управления, так как комбинационные схемы этих систем в основном строятся с применением электромагнитных реле типов РЭС-47, РЭС-48, РЭС-9, имеющих по два переключающих контакта. При использовании таких реле необходимо решить следующий вопрос: как объединять вторые контакты, если первые объединены в один из рассмотренных выше настраиваемых модулей?

Проблема использования вторых контактов решается достаточно просто в случае, если при построении модулей применяется поэлементное резервирование. В системах с поэлементным резервированием против обоих видов отказов (незамыкание и неразмыкание) замыкающие контакты одного реле соединяют последовательно (они могут рассматриваться функционально как один контакт), в то время как контакты разных реле соединяют параллельно.

Предположим, что поэлементное резервирование в модулях не применяется. Тогда, если первые контакты различных реле объединены в цепочку, то объединять вторые контакты этих реле в цепочку (даже другую) не рекомендуется, так как при этом обычно не удается обеспечить «проходимость» схемы второй цепочкой. Вторые контакты реле могут объединяться в структуру типа «звезда», так как использование этого типа модулей не ухудшает «проходимости» схемы цепочкой, состоящей из первых контактов.

Третьим вопросом, определяющим эффективность применения предлагаемых решений, является вопрос о влиянии способа «раздачи» питания на структуру настраиваемых модулей. При децентрализованном построении логических устройств обычно используются различные источники питания, к которым должны подключаться обмотки реле. При такой организации «раздачи» питания возникает проблема: каким образом должны быть объединены обмотки реле при построении модулей?

Эта проблема может быть решена, если отказаться от объединения обмоток звездой. Однако это приводит к существенному возрастанию числа внешних выводов и усложнению монтажа. Поэтому в этом случае рекомендуется применять объединение обмоток небольшими группами из 1-6 реле, а контакты, принадлежащие этой группе реле, соединять в цепочку.

Из сказанного следует, что для повышения уровня интеграции целесообразно применять структуры с централизованной обработкой

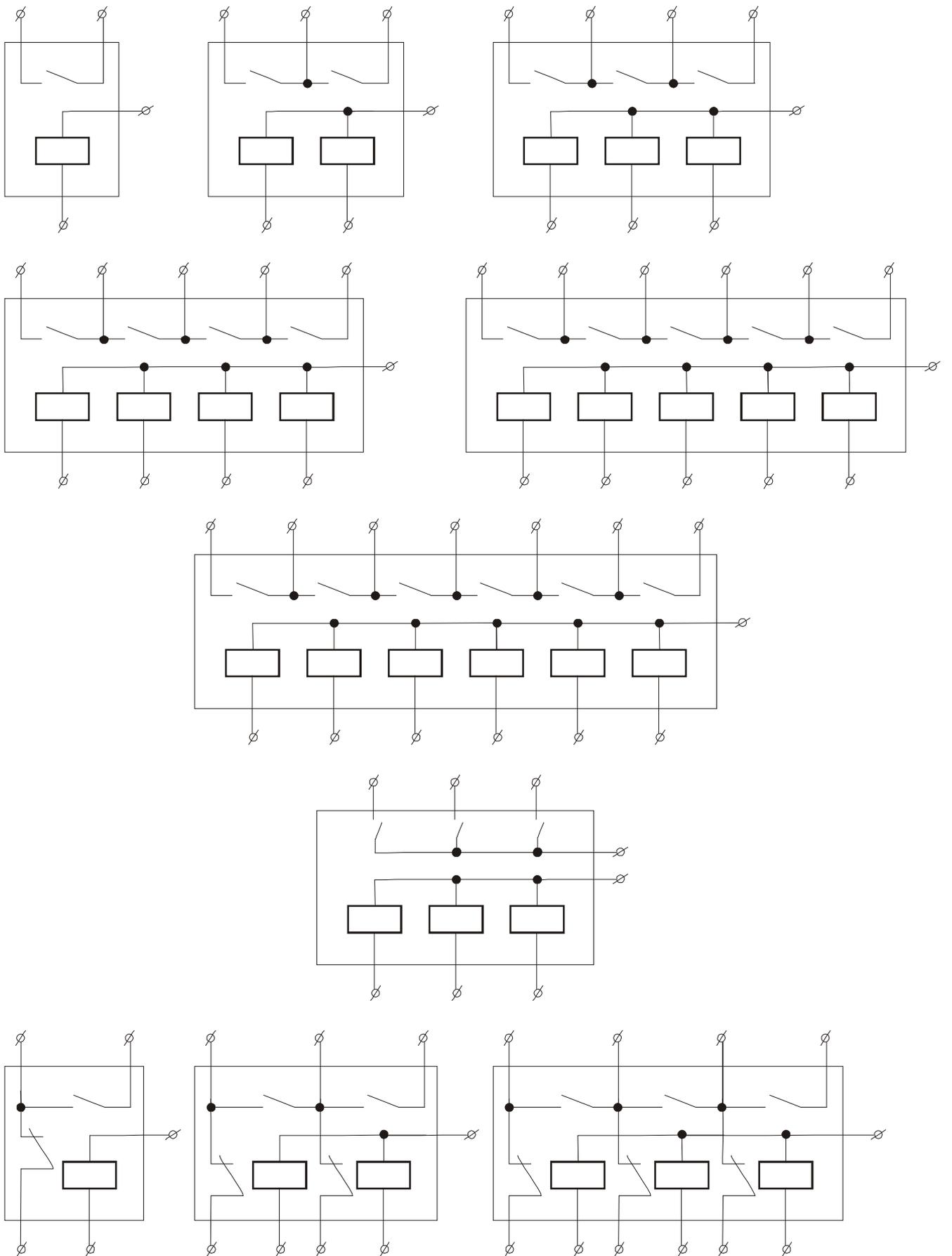


Рис. 7.16. Набор типовых модулей из одноконтактных реле для формирования унифицированных логических кассет

информации и одним источником питания. При этом все обмотки реле могут быть объединены по одному полюсу.

При выборе номенклатуры модулей для формирования унифицированных кассет, предназначенных для реализации нерегулярных логических схем, следует применять возможно более

длинные цепочки, так как это позволяет при заданном числе внешних выводов кассет повысить их функциональные возможности. Однако длина цепочек обычно ограничивается, ввиду того что при реализации схемы, содержащей большое число контактов в базисе единой цепочки, внесение изменений потребует существенной перепайки.

Сокращение длины цепочек ($l \leq 6$) связано также и с тем, что для изоляции фрагментов длинной цепочки приходится не использовать контакты некоторых реле, что приводит к избыточности. Для уменьшения этой избыточности длина цепочек также должна быть сокращена. Опыт проектирования систем логического управления показывает, что в составе унифицированных кассет должны использоваться цепочки длиной 1, 2, 3, 4, 5 и 6. В кассеты должны входить также модули, имеющие структуру типа «звезда», и модули из замыкающих и размыкающих контактов (рис. 7-16).

Заканчивая эту книгу, мы понимаем, что в ней не удалось изложить все вопросы построения и применения настраиваемых логических модулей. Практически остались вне рассмотрения вопросы обеспечения эксплуатации и обслуживания систем на НЛМ, а также все, что связано с НЛМ последовательного и параллельно-последовательного типов.

Однако мы и не ставили своей целью «закрыть» проблему. После выполнения исследований число вопросов обычно возрастает, но это уже вопросы следующего уровня. Недаром древние мудрецы сравнивали знания с кругом света от фонаря: чем шире круг, тем больше граница с темнотой. Поэтому сейчас мы останавливаемся. То, что изложено в этой книге, помогло нам делать системы лучше и проще; мы надеемся, это принесет пользу и читателю. Но впереди еще масса новых нерешенных задач - своих в каждой области автоматизации и общих, направленных на дальнейшее повышение качества и эффективности систем и устройств. Если настоящая книга поможет в их решении, то мы будем считать свою цель достигнутой.

СПИСОК ЛИТЕРАТУРЫ

1. Автоматическое управление газотурбинными установками/Б. И. Аранович, Ю. Т. Лячек, В. А. Олейников, А. А. Файнштейн. - Л.: Недра, 1974.-216 с., ил.
2. Алгоритм построения логической схемы в базисе «И - НЕ» по заданной формуле/В. Л. Артюхов, В. К. Бородулин, Ю. В. Каталажнов, А. А. Шалыто.- Обмен опытом в радиопромышленности, 1973, № 10, с. 23-28.
3. Аперiodические автоматы/Под ред. В. И. Варшавского. - М.: Наука,1976.- 424 с., ил.
4. Артюхов В. Л., Копейкин Г. А., Шалыто А. А. Оценки сложности предстоящих реализаций в задачах логического и технического проектирования.-В кн.: Теория релейных устройств. Труды XVI Всесоюзной школы-семинара. - Челябинск: ЧПИ, 1976, с. 123-129.
5. Артюхов В. Л., Копейкин Г. А., Шалыто А. А. Применение многофункциональных модулей среднего уровня интеграции для построения комбинационных схем. - Теория конечных автоматов и ее приложения, 1976, вып. 5,с. 86-110.
6. Букреев И. Н., Мансуров Б. М., Горячев В. И. Микроэлектронные схемы цифровых устройств. - М.: Советское радио, 1975. - 368 с., ил.
7. Бутаков Е. А. Методы синтеза релейных устройств из пороговых элементов.-М.: Энергия, 1970.- 328 с., ил.
8. Варшавский В. И., Мараховский В. Б., Песчанский В. А. Реализация инвариантных функций линейными однородными схемами. - Изв. АН СССР.Сер. техн. кибернетика, 1969, № 4, с. 39-44.
9. Варшавский В. И., Мараховский В. Б., Песчанский В. А. Многофункциональные модули, реализующие все функции трех и четырех переменных.-В кн.: Тезисы докладов II Всесоюзного совещания по теории релейных устройств и конечных автоматов. - Рига: Зинатне, 1971, с. 53-55.
- 10.Васильев Ю. Л. Минимальные контактные схемы для булевых функций четырех переменных. - Докл. АН СССР, 1959, № 2, с. 32-36.
- 11.Вольский В. Е., Пушин Ю. Н., Юнг В. Н. Проектирование пневматических систем управления судовыми энергетическими установками. - Л.:Судостроение, 1975.- 312 с., ил.
- 12.Выбор базиса элемента одномерной вычислительной среды/А. А. Койфман, Б. А. Москов, А. В. Панков и др. - Вычислительные системы, 1973,вып. 54, с. 53-66.
- 13.Гаврилов М. А. Теория релейно-контактных схем.- М. -Л.: Изд-во АН СССР, 1950.- 304 с., ил.
- 14.Гаврилов М. А., Копыленко В. М. Метод «переходных таблиц» синтеза многовыходных комбинационных структур на произвольных элементах.-В кн.: Абстрактная и структурная теория релейных устройств. - М: Наука,1972, с. 57-76.
- 15.Горбатов В. А. Синтез логических схем в произвольном базисе.-В кн.: Теория дискретных автоматов. - Рига: Зинатне, 1967, с. 73-84.
- 16.Данем Б., Норт Д. Проблемы выбора логических эффективных основных ячеек. - В кн.: Синтез релейных структур. - М.: Наука, 1965, с. 108-122.
17. Диденко К. И., Загарий Г. И., Павлов В. В. Управляющие логические устройства и их применение в АСУ. - Приборы и системы управления, 1975,№ 8, с. 3-8.
18. Дулепов Е. Г. Функциональные возможности универсальных симметричных логических элементов. - Автоматика и телемеханика, 1974, № 4,с. 83-87.
- 19.Евреинов Э. В., Прангишвили И. В. Цифровые автоматы с настраиваемой структурой. - М.: Энергия, 1974.- 240 с., ил.

20. Захаров В. Н. Автоматы с распределенной памятью. - М.: Энергия, 1975.- 136 с., ил.
21. Захаров В. Н., Поспелов Д. А., Хазацкий В. Е. Системы управления. Задание. Проектирование. Реализация. - М.: Энергия, 1977. - 423 с., ил.
22. Зыков А. А. Теория конечных графов. - Новосибирск: Наука, 1969.-543 с., ил.
23. Карпова Н. А. (Минимальные схемы из замыкающих контактов для монотонных функций пяти переменных. – Проблемы кибернетики, 1973, вып. 26, с. 45-84.
24. К'огаи И. В., Стеоликовья Л. Н. Вопросы построения полных тестов для бесповторных скобочных формул. - В кн.: Техническая диагностика.-М.: Наука, Г9/2, с 9/--105.
25. Кондратьева Е. А. Сопруненко Е. П. Об оценке числа входов универсального логического модуля. - В кн.: Автоматы и управление сетями связи.- М.: Наука, 1971, с. 20-24.
26. Кукинов А. М. Простой метод синтеза скобочных формул для недоопределенных булевых функции. - В кн.: Синтез дискретных автоматов и управляющих устройств. - М.: Наука, 1968, с. 118-124.
27. Лазарев В. Г., Пийль Е. И. Синтез асинхронных конечных автоматов. - М.: Энергия, 1970.- 400 с., ил.
28. Лофгрэн Л. Теория однородных переключательных сетей. - В кн.: Синтез релейных структур. - М.: Наука, 1965, с. 142-176.
29. Лупанов О. Б. О синтезе некоторых классов управляющих систем.-Проблемы кибернетики, 1963, вып. 10, с. 63-97.
30. Майоров С. А., Скорубский В. И., Кравцов Л. Я. Универсальные логические модули и их применение. - Управляющие системы и машины, 1977, № 1. с. 27-37.
31. Мелихов А. Н., Бернштейн Л. С., Курейчик В. Н. Применение графов для проектирования дискретных устройств. - М.: Наука, 1974.-303 с., ил.
32. Микроэлектроника и однородные структуры для построения логических и вычислительных структур/И. В. Прангишвили, Н. А. Абрамова, Е. В. Бабичева, В. В. Игнатущенко. - М.: Наука, 1974.- 228 с., ил.
33. Миллер Р. Теория переключательных схем. Т. 1. - М.: Наука, 1970.-416 с., ил.
34. Некоторые критерии оценки эффективности логических модулей/Е. П. Бабичева, И. В. Прангишвили, М. А. Ускач, Н. Шаипов. – Автоматика и телемеханика, 1968, № 3, с. 131-139.
35. Образцов В. В. Синтез устройства программирования работы агрегатов при отключении их по одному. - В кн.: Автоматизация управления оросительными системами -Фрунзе: Илим, 1970, с. 53-59.
36. Оверко В. А. О результатах анализа алгоритмов решения класса логических промышленных задач. - В кн.: Надежность и устройства автоматизированных систем управления. - Киев: Наукова думка, 1970, с. 33-39.
37. О выборе эффективных булевых функций/С. А. Майоров, В. В. Павленко, Г. А. Петухов, В. И. Скорубский. - Изв. вузов. Сер. приборостроение, 1970, № 1, с. 34-40.
38. Однородные структуры/ В. И. Варшавский, В. Б. Мараховский, В. А. Песчанский, Л. Я. Розенблум. - М.: Энергия, 1973.- 152 с., ил.
39. О переписи типовых релейно-контактных схем автоматической телефонии/В. Г. Лазарев, О. Ф. Наумчук, Г. Г. Саввин, Ю. Л. Сагалович. - Проблемы передачи информации, 1963, № 8, с. 63-69.
40. Павлов В. В., Никитин А. В. Логические блоки для управления исполнительными механизмами. - М.: Энергия, 1972.- 64 с., ил.
41. Поваров Г. Н. Метод синтеза вычислительных и управляющих контактных схем. - Автоматика и телемеханика, 1957, № 2, с. 55-59.
42. Попов Ю. А., Бочков П. Е., Сладкое А. Б. Комплект многофункциональных модулей для построения вычислительных устройств. - Приборы и системы управления, 1973, № Ц, с. 14-17.
43. Поспелов Д. А. Логические методы анализа и синтеза схем. - М.: Энергия, 1968.- 328 с., ил.
44. Прангишвили И. В., Ускач М. А., Копейкип Г. А. Комплекс логических МДП-интегральных схем для систем автоматики и телемеханики.-Приборы и системы управления, 1970, № 4, с. 15-18.
45. Проектирование бесконтактных управляющих логических устройств промышленной автоматики/Г. Р. Грейнер, В. П. Ильяшенко, В. П. Май и др.-М.: Энергия, 1977.- 384 с., ил.
46. Проектирование цифровых вычислительных машин/С. А. Майоров, Г. И. Новиков, С. И. Баранов и др. - М.: Высшая школа, 1972.-300 с., ил.
47. Расчет и проектирование элементов ЭВМ/Ю. А. Попов, Б. И. Кальнин, Л. С. Соловьев и др. - М.: Атомиздат, 1975. -310 с., ил.

- 48.Редькин Н. П. О сложности реализации недоопределенных булевых функций. - Автоматика и телемеханика, 1969, № 9, с. 42-47.
- 49.Риордан Дж. Введение в комбинаторный анализ. - М.: Изд-во иностр. лит., 1963. - 287 с., ил.
- 50.SHP-программа стандартизации радиоэлектронной аппаратуры ВМС США/А. С. Бейлин, И. М. Гаврилов, В. В. Громковский, Н. М. Гусев. - Судостроение за рубежом, 1976, № 9, с. 35-48.
- 51.Справочник по интегральным микросхемам/Под общей ред. Б. В. Тарабрина.-М.: Энергия, 1977. - 583 с., ил.
- 52.Стародубцев Н. А. Соотношения для числа настроек многофункциональных логических модулей. - Изв. АН СССР. Сер. техн. кибернетика, 1972, №4, с. 145-152.
- 53.Хетагуров Я. А., Малишевский В. В., Потураев О. С. Основы инженерного проектирования УЦВМ. - М.: Советское радио, 1972. - 367 с., ил.
- 54.Централизованное управление общесудовыми системами/В. М. Дунин, В. Л. Артюхов, А. Г. Попов и др. - Л.: Судостроение, 1976. - 272 с., ил.
- 55.Цымбал А. А., Сидоренко В. П., Петин Ю. А. Цифровые интегральные схемы на МДП-структурах. - Электронная промышленность, 1974, № 12, с. 57-64.
- 56.Чистов В. П., Битюцкий В. П. Функциональная полнота в ленточных однородных структурах. - Изв. АН СССР. Сер. техн. кибернетика, 1971, №3, с. 47-53.
- 57.Шеннон К. Э. Работы по теории информации и кибернетике. - М.:Изд-во иностр. лит., 1963.- 829 с., ил.
- 58.Яблонский С. В. Функциональные построения в /С-значной логике.-Тр. Математического института им. В. А. Стеклова, т. 51, 1958, с. 5-142.
- 59.Якубайтис Э. А. Логические автоматы и микромодули. - Рига: Зинатне, 1975. - 259 с., ил.
- 60.Cohen S., Winder R. Threshold gate building blocks. - IEEE Trans.Comput., 1969, № 9, p. 32-42.
- 61.Forslund p. C., Waxman R. The universal logic block (ULB) and its application to logic design. - In: Proc. 7th IEEE Ann. Symp. Switching and Automata Theory, 1966, p. 15-23.
- 62.Hellerman L. A catalogue of three-variable or-invert and and-invert logical circuits. - IEEE Trans. Comput., 1963, N 12, p. 3-15.
- 63.Hurst S. Universal threshold gates for 3, 4, 5 variable linearly separable functions. -Electr. Lett., 1970, № 6, n. 22-32.
- 64.Ikeno N., Hashimoto A., Naito K. A table of fourvariable minimal NAND circuits. - Electr. Comm. Lab. Tech. J., extra issue, 1968, № 26, p. 92-100.
- 65.King W. F. The synthesis of multipurpose logic devices. - In: Proc.7th IEEE Ann. Symp. Switching and Automata Theory, 1966, p. 78-85.
- 66.Levy S. Y., Winder R., Mott T. H. A note on tributary switching networks. - IEEE Trans. Comput., 1964, № 2, p. 36-45.
- 67.Maitra K. K. Cascaded switching networks of two-input flexible cells.-IRE Trans. El. Comput., 1964, № 2, p. 25-31.
- 68.Makhapadhyay A. Unate cellular logic.- IEEE Trans. Comput., 1969, № 2, p. 18-27.
69. Recent developments in switching theory/Ed, by A. Makhapadhyay. -New York: Acad. Press, 1971. -583 p.
- 70 Osman M. Y., Weiss C. D. Universal base functions and modules for realizing arbitrary switching functions. - IEEE Trans. Comput., 1972, № 9, p. 114-127.
71. Patt Y. N. A complex module for the synthesis of combinational switching circuits.- In: Proc. 1967 AFIPS Spring Joint Comput. Conf. 20, p. 72-85.
- 72.Sklansky J., Korenjak A. J., Sone H. S. Canonical tributary networks. - IEEE Trans. Comput., 1965, A^rs 6, p. 62-68.
- 73.Smith R. A. Minimal three variable NOR and NAND logic circuits. -ИГ 1-E Trans. Comput., 1969, № 2, p. 114-117.
- 74.Yau S. S., Tang C. K. Universal logic modules and their applications.-11.1 T: Trans. Comput., 1970, № 2, p. 18-23.

Введение.....	3
Глава первая. Логическое управление в системах автоматики.....	10
1-1. Задачи логического управления. Структура и техническая реализация логических устройств	-
1-2. Способы задания алгоритмов работы управляющих логических устройств.....	13
Глава вторая. Функциональный базис настраиваемых модулей	19
2-1. Настраиваемые модули. Основные понятия и определения	-
2-2. Выбор функционального базиса модулей	22
2-3. Свойства неповторных булевых формул	30
Глава третья. Настраиваемые модули из функциональных элементов	42
3-1. Порождающие функции. Свойства и характеристики-	
3-2. Построение модулей с минимальным числом внешних выводов.....	47
3-3. Пример построения порождающей функции настраиваемого модуля.....	61
3-4. Техническая реализация настраиваемых логических модулей.....	68
Глава четвертая. Построение логических схем из настраиваемых логических модулей.....	71
4-1. Реализация схем в базисе настраиваемых модулей	-
4-2. Оценки сложности реализации	84
Глава пятая. Цифровые интегральные микросхемы широкого применения как настраиваемые модули	95
5-1. Оценка логической эффективности микросхем.....-	
5-2. Построение логических схем на микросхемах.....	101
5-3. Унификация блоков логики, построенных из микросхем широкого применения.....	106
Глава шестая. Однородные структуры из настраиваемых модулей	114
6-1. Однородные структуры. Основные понятия и определения	-
6-2. Реализация булевых формул в однородных структурах из комбинационных элементов	116
Глава седьмая. Настраиваемые модули из элементов с двусторонней проводимостью.....	126
7-1. Особенности схем из элементов с двусторонней проводимостью.....-	
7-2. Выбор структуры настраиваемых модулей из ЭДП.....	129
7-3. Построение схем из ЭДП в базисе настраиваемых модулей	133
7-4. Унификация блоков логики из элементов с двусторонней проводимостью	153
Заключение	159
Список литературы.....	160

Валерий Леонидович Артюхов

Геннадий Александрович

Копейкин Анатолий Абрамович Шалыто

НАСТРАИВАЕМЫЕ МОДУЛИ ДЛЯ УПРАВЛЯЮЩИХ ЛОГИЧЕСКИХ УСТРОЙСТВ

Редактор *В. А. Романов*

Художественный редактор *Д. Р. Стеванович*

Технический редактор *А. Г. Рябкина*

Корректор *В. В. Румянцев*

Обложка художника *Ю. И. Прошлецова*

ИБ № 1153 („Энергия“)

Сдано в набор 27.01.81. Подписано в печать 14.05.81. М-21484. Формат 60X90 $\frac{1}{16}$ - Бумага типографская № 2. Гарнитура литературная. Печать высокая. Усл. печ. л. 10,5. Уч.-изд. л. 11,4. тираж 7000 экз. Заказ 978. Цена 60 к.

Ленинградское отделение Энергоиздата. 191041, Ленинград, Марсово поле, 1.

Ленинградская типография № 2 ч головное предприятие ордена Трудового Красного Знамени Ленинградского объединения «Техническая книга» им. Евгении Соколовой Союзполиграфпрома при Государственном комитете СССР по делам издательств, полиграфии и книжной торговли, .198052, Ленинград, Измайловский проспект, 29,

В КНИГОТОРГАХ СТРАНЫ ИМЕЮТСЯ СЛЕДУЮЩИЕ КНИГИ ИЗДАТЕЛЬСТВА «ЭНЕРГИЯ»

В Волгоградском

Брынский Е. А., Данмлевич Я. Б., Яковлев В. И. Электромагнитные поля в электрических машинах. Л., Энергия, 1979. 80 к. Имеется 10 экз.

Извозчиков В. А., Тимофеев О. А. Фотопроводящие окислы свинца в электронике. Л., Энергия, 1979. 65 к. Имеется 10 экз.

400001, Волгоград, Рабоче-Крестьянская ул., 13. С/с 92453102 в городском управлении Госбанка.

В Донецком

Захаров В. К. Электронные элементы автоматики. Учебник для вузов по специальности «Автоматика и телемеханика».- 2-е изд., перераб. и доп. Л., Энергия, 1975. 1 р. 05 к. Имеется 90 экз.

340002, Донецк, ул. Б. Хмельницкого, 102. С/с 92453001 в Калининском отделении Госбанка.

В Кзыл-Ординском

Домбровский В. В., Хуторецкий Г. М. Основы проектирования электрических машин переменного тока. Л., Энергия, 1974. 2 р. 43 к. Имеется 56 экз.

460701, Кзыл-Орда, ул. 60 лет Октября, 143. С/с 9245302 в городском отделении Госбанка.

В Киевском

Заездный А. М. Гармонический

синтез в радиотехнике и электросвязи.- 2-е изд., испр. и доп. Л., Энергия, 1972. 1 р. 96 к. Имеется 121 экз.

Цифровые схемы распределения тока. Л., Энергия, 1969.

70 к. Имеется 273 экз.

252124, Киев, ул. Радищева, 12/16. С/с 92045305, в областной конторе Госбанка.

В Кировском

Использование водной энергии. Учеб. пособие для втузов/ Д. С. Щавелев, П. И. Васильев, Ю. С. Васильев и др. Под ред. Д. С. Щавелева. Л., Энергия, 1976. 2 р. 46 к. Имеется 10 экз. 610000, Киров, ул. Ленина, 88. С/с 9245303 в городском управлении Госбанка.

В Ленинадском

Меерсон А. М. Радиоизмерительная техника. -3-е изд. пере-раб. и доп. Л., Энергия, 1978. Массовая радиобиблиотека (Вып. 960). 2 р. 50 к. Имеется 60 экз. 735690, Ленинад, Кайраккумский поворот. С/с 9245302 в городском отделении Госбанка.

В Мособлкниге

Анисимов В. И. Топологический расчет электронных схем. Л., Энергия, 1977. 1 р. 10 к. Имеется 70 экз. 111399, Москва, ул. Metallургов, 58. С/с 9245202 в областной конторе Госбанка.

В Новочеркасском

Дельвинг Г. Н., Траскунов П. М., Царюк Н. М. Управление качеством продукции в электроприборостроении. Л., Энергия, 1977. 70 к. Имеется 31 экз.

Иоссель Ю. Я. Расчет потенциальных полей в энергетике. Справочная книга. Л., Энергия, 1978. 1 р. 30 к. Имеется 30 экз.

346430, Новочеркасск, пр. Ленина, 10. С/с 92453303 в областной конторе Госбанка.

В Павлодарском

Авербух А. М. Релейная защита в задачах с решениями и примерами. Л., Энергия, 1975. 1 р. 43 к. Имеется 95 экз.

637021, Павлодар, ул. Фрунзе, 151. С/с 9245301 в городской конторе Госбанка.

В Свердловском

Пехович А. И., Жидких В. М. Расчеты теплового режима твердых тел. -2-е изд., перераб. и доп. Л., Энергия, 1976. 1 р. 32 к. Имеется 34 экз.

620046, Свердловск, Артинская ул., 23-А. С/с 92453701 в городском управлении Госбанка.